

Review

A Comprehensive Review of End-to-End Autonomous Driving: Architectures and Emerging Trends

Yunxing Chen ^{1,2}, Guo Yu ^{1,2,*} , Pengfei Ran ^{1,2} and Zhijun Chen ³¹ Hubei Key Laboratory of Vehicle-Infrastructure Collaboration and Traffic Control, Hubei University of Arts and Science, Xiangyang 441053, China; chenyunxing@chd.edu.cn (Y.C.); 202508551062@hbuas.edu.cn (P.R.)² School of Automobile and Traffic Engineering, Hubei University of Arts and Science, Xiangyang 441053, China³ The Intelligent Transportation Systems Research Center, School of Transportation and Logistics Engineering, Wuhan University of Technology, Wuhan 430061, China; chenzj556@whut.edu.cn

* Correspondence: 202408551037@hbuas.edu.cn

Abstract

End-to-end autonomous driving is an emerging technology and a prominent research focus in both industry and academia. By integrating perception, localization, decision-making, and control into a single model, end-to-end systems aim to streamline the traditional modular pipeline while introducing new challenges in safety validation and interpretability. Unlike existing surveys that predominantly catalog algorithms, this review proposes a novel function-oriented taxonomy by categorizing architectures into perception-integrated and planning-integrated paradigms. Beyond the architectural dimension, the analysis delves into critical safety and interpretability, emphasizing the fundamental gap between theoretical design and the reliability required for real-world deployment. Industrial applicability is examined through real-world examples of data closed-loop workflows and simulation testing, addressing practical constraints in latency and computing resources. Finally, the review addresses critical challenges, particularly long-tail data scarcity and the deficiency in human-like decision-making and outlines future directions toward achieving robust autonomy.

Keywords: autonomous driving; end-to-end; deep learning; explainability; generalization

1. Introduction

With the rapid increase in the number of motor vehicles world-wide, road accidents and traffic congestion are becoming increasingly prominent. Studies have found that human driver errors account for more than 90% of the causes of total traffic accidents [1]. This shows that the risks posed by human drivers while driving cannot be ignored. At the same time, the rapid intelligent transformation of the automotive industry has put forward a strong industrial demand for autonomous driving technology. Companies such as Tesla and XPeng have actively deployed autonomous driving technology to gain an advantage in future competition. In addition, governments and relevant departments have introduced supportive policies to promote the advancement of autonomous driving technology. This shows that the transition of drivers from human operators to machine systems has become an obvious trend.

Autonomous driving systems can be broadly categorized into two primary approaches: (i) modular systems and (ii) end-to-end systems. Modular systems achieve autonomous



Academic Editors: Stefano Coropulis and Md Abdus Samad Kamal

Received: 18 April 2026

Revised: 11 July 2026

Accepted: 22 July 2026

Published: 6 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

driving tasks through a staged pipeline that includes perception, localization, decision-making, and control. During the early stages of autonomous driving, most systems adopted modular architectures, which offered significant advantages in terms of interpretability and ease of debugging for individual modules. However, as driving scenarios have grown increasingly complex, the limitations of modular architectures have become more pronounced. Notably, challenges such as error propagation between modules and limited adaptability to dynamic environments hinder the overall performance of modular approaches. First, the perception module may misclassify objects due to variations in lighting conditions (e.g., glare or shadows), and this error can be exacerbated as it propagates to the decision-making module, ultimately compromising the safety and reliability of the control module's outputs [2]. To address these limitations, end-to-end autonomous driving systems [3–6] have emerged as a promising alternative. Second, due to the obvious timing lag between modules in the modular architecture, response delays may occur in complex dynamic scenarios, affecting the safety of the system.

The end-to-end approach directly maps sensor inputs to vehicle control outputs using a single model, with one of the most commonly employed techniques being Convolutional Neural Networks (CNNs) [3]. This supervised learning paradigm offers two major advantages: it eliminates the issue of error accumulation caused by message passing between modules in modular systems and enables effective training on large-scale driving datasets. Additionally, end-to-end autonomous driving leverages supervised learning to pretrain raw input data and generate future scenario predictions, thereby achieving a data-driven framework [7]. However, this approach, which directly maps inputs to outputs, poses a significant drawback: poor interpretability. In other words, understanding the rationale behind incorrect model behaviors remains challenging. This lack of interpretability has led researchers to focus on enhancing the transparency of models. For instance, integrating Transformer architectures [8,9] with attention visualization techniques [10,11] or incorporating multimodal fusion [12,13] has shown promising results in improving interpretability.

In recent years, researchers have conducted a series of outstanding studies on end-to-end autonomous driving. For example, Tampuu et al. [2] summarized early E2E driving architectures and training methods. Chen et al. [14] offer a forward-looking exploration of emerging challenges and frontiers. Singh et al. [15] discuss a deep learning-based end-to-end autonomous driving method from methodological and technical perspectives. Ganesan et al. [16] focused on DL-based motion planning and E2EL; Zhao et al. [17] provided a system-level survey of autonomous driving; Hu et al. [18] reviewed end-to-end perception and prediction; Zhao et al. [19] surveyed autonomous driving from a deep learning perspective; Yang et al. [20] unified conventional, VLM-centric, and hybrid E2E paradigms; and Dong et al. [21] discussed the transition from classic E2E paradigms to large model-empowered autonomous driving. In contrast, our survey covers the latest developments and a wide range of topics. A detailed comparison is provided in Table 1.

Motivated by these gaps, this review is guided by three central questions concerning: (1) the function-oriented distinction of end-to-end architectures in terms of output interfaces, intermediate planning representations, and downstream controller coupling; (2) the influence of learning paradigms and architectural choices on safety, interpretability, and evaluation reliability; and (3) the key challenges and future directions for improving robustness, generalization, and deployability, particularly when end-to-end systems interface with actuation modules.

Table 1. Comparison with representative surveys on autonomous driving and end-to-end autonomous driving.

Survey	Topic Focus	Taxonomy and Perspective	Main Contribution
Tampuu et al. [2]	E2E driving architectures and training methods	It organizes E2E driving around learning paradigms, input–output modalities, network architectures, evaluation schemes, interpretability, and safety, providing an early systematic overview of model design choices.	Provides an early systematic overview of E2E driving from the perspective of model design choices and training strategies.
Chen et al. [14]	Challenges and frontiers of end-to-end autonomous driving	This survey is organized around the motivation, roadmap, methodology, benchmarks, challenges, and future trends of E2E driving, with emphasis on multi-modality, visual abstraction, world modeling, interpretability, safety, causal confusion, robustness, data engines, and foundation models.	Provides a frontier-oriented synthesis of E2E autonomous driving and highlights open challenges that shape later research directions.
Singh [15]	Deep learning-based E2E autonomous driving	Frames E2E driving as a deep learning stack and reviews input modalities, output modalities, learning approaches, domain adaptation, safety, explainability, evaluation, datasets, and simulators.	Provides a component-level taxonomy of recent E2E methods, especially from the perspectives of modality, learning principle, safety, explainability, and evaluation.
Ganesan et al. [16]	Deep learning-based motion planning and E2E learning for self-driving vehicles	Centers on motion planning and organizes the literature into behavior planning, trajectory planning, E2EL, imitation learning, reinforcement learning, datasets, simulation frameworks, performance evaluation, and implementation analysis.	Connects DL-based behavior/trajectory planning with E2EL and emphasizes practical enablers such as datasets, simulation, deployment frameworks, and implementation issues.
Zhao et al. [17]	General autonomous driving systems	Adopts a system-level ADS organization, covering layered and E2E architectures, scene perception, localization, motion planning, decision-making, simulators, scenario generation, challenges, and outlook.	Provides a broad overview of autonomous driving systems rather than an E2E-specific taxonomy, positioning E2E as one architecture among wider ADS components.
Hu et al. [18]	End-to-end perception and prediction for autonomous driving	Focuses on the upstream pipeline and organizes methods by task formulation, datasets, metrics, LiDAR/camera/multimodal encoding, decoding strategies, computational cost, and output forms such as trajectories and intentions.	Provides a focused survey of joint perception prediction integration, especially sensor encoding and prediction output design before planning and control.
Zhao et al. [19]	Autonomous driving from a deep learning perspective	Reviews autonomous driving through DL-enabled modules, including perception and detection, localization and mapping, decision-making and control, efficiency, real-world benchmarks, safety, and explainability.	Provides a broad DL-for-AD synthesis across multiple modules, with E2E decision-making treated as one part of a larger deep learning-driven autonomous driving landscape.
Yang et al. [20]	General end-to-end autonomous driving	Proposes a GE2E perspective and organizes recent methods into conventional E2E, VLM-centric E2E, and hybrid E2E paradigms, together with datasets, benchmarks, performance comparison, challenges, and future trends.	Provides a unified view of conventional E2E, VLM-centric E2E, and hybrid E2E systems, emphasizing the convergence between vision-centric models and vision–language reasoning.
Dong et al. [21]	E2E autonomous driving from classic paradigms to large-model empowerment	Discusses the evolution from classic E2E paradigms to large model-empowered E2E systems, including learning paradigms, foundation models, reasoning, world models, data generation, scene understanding, and planning.	Positions recent E2E autonomous driving within the transition from imitation/reinforcement learning toward large model-empowered perception, reasoning, and planning.
Ours	End-to-end autonomous driving architectures, learning paradigms, safety, interpretability, benchmarks, deployment	Organizes E2E autonomous driving using a function-oriented taxonomy that separates perception–control integration from perception–planning integration according to output type, intermediate planning representation, and coupling with downstream control modules.	Provides a function-oriented synthesis that links E2E architecture design with planning representations, vehicle control, actuator-level execution, deployment constraints, safety, and interpretability.

We collected and screened more than 190 closely related publications on end-to-end autonomous driving from major academic databases and repositories, including IEEE Xplore, Web of Science, ScienceDirect, SpringerLink, and arXiv. The literature search primarily focused on publications from 2014 to 2025, while several earlier foundational studies were also included when necessary to provide historical context. The search used keywords including end-to-end autonomous driving, trajectory planning, imitation learning, reinforcement learning, Bird's-Eye View (BEV) representation, Transformer autonomous driving, multi-modal fusion, autonomous driving benchmarks and world model. The included studies were required to be directly related to end-to-end or learning-based autonomous driving architectures. The main contributions of this work can be summarized as follows:

- We propose a mutually exclusive and clear function-oriented classification framework for end-to-end architectures, which addresses the problem of easy overlap in traditional end-to-end and modular end-to-end frameworks, and we discuss the current practical applications of end-to-end technologies.
- We propose a systematic correspondence between current challenges and future trends. Specifically, we address data scarcity through cross-scenario learning and digital twins, bridge the gap in human-like decision-making by integrating cognitive mechanisms, and enhance interpretability through reasoning-driven spatial understanding. This logic-driven analysis provides theoretical guidance for overcoming current challenges.
- Addressing the disconnect between academic research and real-world applications, we have reviewed representative industry solutions and highlighted that sim-to-real migration and data closed-loop iteration represent core scalability barriers. This analysis guides transitioning from academic research to large-scale deployment.

The rest of this paper is organized as follows: Section 2 first introduces the basic concepts and technical background of end-to-end autonomous driving, systematically compares the characteristics of modular architecture and end-to-end architecture, and proposes a clear classification method for models; Section 3 deeply explores the key technologies and paradigms of end-to-end autonomous driving systems, including key application features and technical details such as imitation learning, reinforcement learning, simulator to real, Transformer architecture and world model; Sections 4 and 5 specifically discuss the safety and explainability issues of end-to-end autonomous driving systems and review the relevant research progress; Section 6 analyzes in detail the evaluation methods of end-to-end autonomous driving systems, clarifies the advantages and disadvantages of open-loop and closed-loop evaluation, and summarizes the current mainstream datasets and simulation platforms; Section 7 summarizes the current application status of end-to-end autonomous driving in the industry; Section 8 systematically summarizes the core challenges currently faced by end-to-end autonomous driving systems, and proposes forward-looking research directions for these challenges, including scenario migration and internationalization, cognitive mechanisms, digital twin simulation environments and spatial understanding. Finally, Section 9 summarizes the key conclusions of this survey.

2. Comparison of Modular and End-to-End Approaches

2.1. Modular Architecture

Modular architectures are often referred to as traditional architectures. In the majority of the extant literature, the concept of “pipeline” is mentioned for modularity (e.g., refs. [2,14,15]). The term “pipeline” serves as a metaphor for describing modular architectures and their workflows, and it can reflect the characteristics of task-solving and data flow in modular architectures. This is because the metaphor of a “pipeline” provides a useful lens through which to understand the design and operation of modular architectures. A typical modular autonomous driving system generally comprises a perception module, a

localization module, a decision module, and a control module, as shown in Figure 1. Each of these modules is developed independently and integrated through a “pipeline.” They interact and collaborate on data to automate the vehicle, which is the traditional architecture. Concurrently, they adhere to the traditional divide-and-conquer engineering principles, as the AI driver relies on modules with recognizable responsibilities [22]. A key advantage of the modular architecture is that it possesses delineated module boundaries, which render the entire system highly interpretable. Each module of the modular architecture will be introduced individually as follows:

- **Perception module:** It acts as the “eyes” and “ears” of the system, responsible for comprehensively understanding the vehicle’s surrounding environment. The processed data is then transmitted to the decision-making module. The tasks of the perception module generally include target detection, path tracking, and scene understanding. Techniques commonly employed in this module include multi-sensor fusion and occupancy maps, which integrate data from multiple sensors to enhance the accuracy and depth of environmental information.
- **Localization module:** Localization is defined as the process of recognizing the vehicle’s pose in the world and the vehicle map subsystem. This process may rely on the input of high-definition maps or GPS data and will inform other aspects of self-driving vehicle operation.
- **Decision-making module:** The decision-making module is tasked with formulating optimal behavioral patterns and pathways based on input from the perception module. For example, refs. [23,24] introduced the application of game theory as a modeling tool, particularly in scenarios involving human–machine interactions and complex multi-vehicle environments, to optimize decision-making processes. Similarly, ref. [25] utilized fuzzy theory to design a controller capable of estimating collision probabilities in simultaneous lane-changing scenarios involving two vehicles. These examples illustrate the diverse tasks performed by the decision-making module in modular architectures. Additionally, other methodologies, such as Bayesian theory, have been proposed. However, these will not be discussed in this particular context.
- **Control module:** It is the execution layer of a modular autonomous driving system and is responsible for translating the target behaviors or trajectories generated by the decision module into specific vehicle control commands, such as acceleration, braking, and steering operations. The prevailing methodologies employed in this context are Proportional–Integral–Derivative (PID) [26] and Model Predictive Control (MPC) [27], both of which are designed to achieve objectives such as lane travel.

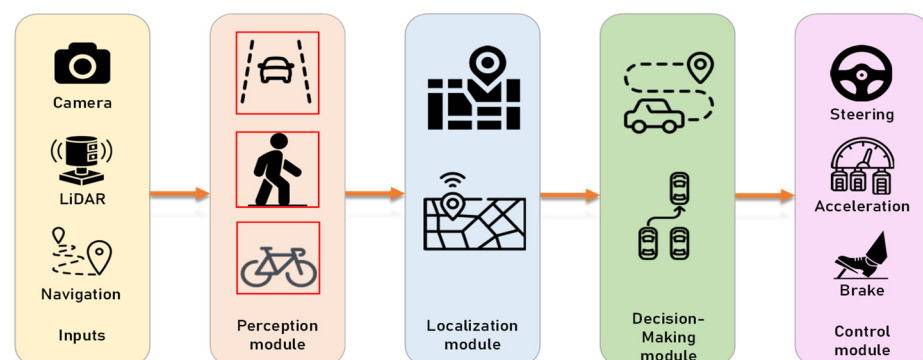


Figure 1. Modular autonomous driving architecture consisting of perception, localization, decision-making, and control modules connected in a sequential pipeline.

Modular architectures, despite their advantages in dealing with environments with clear and predictable rules, have significant limitations. First, since errors introduced by

upstream modules propagate to downstream modules, the accumulation of errors between modules can significantly degrade the overall performance of the system. Second, the modular architecture requires multiple sensors, processors, and algorithms, which tends to lead to less efficient resource utilization, and third, the independence of the modular architecture makes the system relatively less scalable and maintainable.

2.2. End-to-End Architecture

End-to-end autonomous driving is a method of autonomous driving that directly maps sensor inputs into vehicle control commands through a single model. Unlike conventional modular architecture, the end-to-end approach eradicates the staging of processing for individual modules, instead employing a unified model to learn the entire driving task. This approach has been demonstrated to offer the advantage of preventing the accumulation of modular errors and simplifying the data processing pipeline [14].

An end-to-end architecture generally comprises an input modality, a network structure, and an output modality. The concept of end-to-end was first introduced in 1988 by ALVINN [28], who employed a three-layer backpropagation neural network with simulated road images to predict a vehicle's direction of travel based on camera and laser rangefinder inputs. Later, in 2016, NVIDIA [3] directly mapped the raw pixels from a single front-facing camera to the steering commands of a car by training CNNs, enabling the system to autonomously acquire the processing capabilities necessary for driving, including the detection of road features. End-to-end architecture has seen rapid growth since then. In this review, we classify end-to-end architectures according to the inference-stage output interface and its coupling with downstream control modules, as shown in Figure 2.

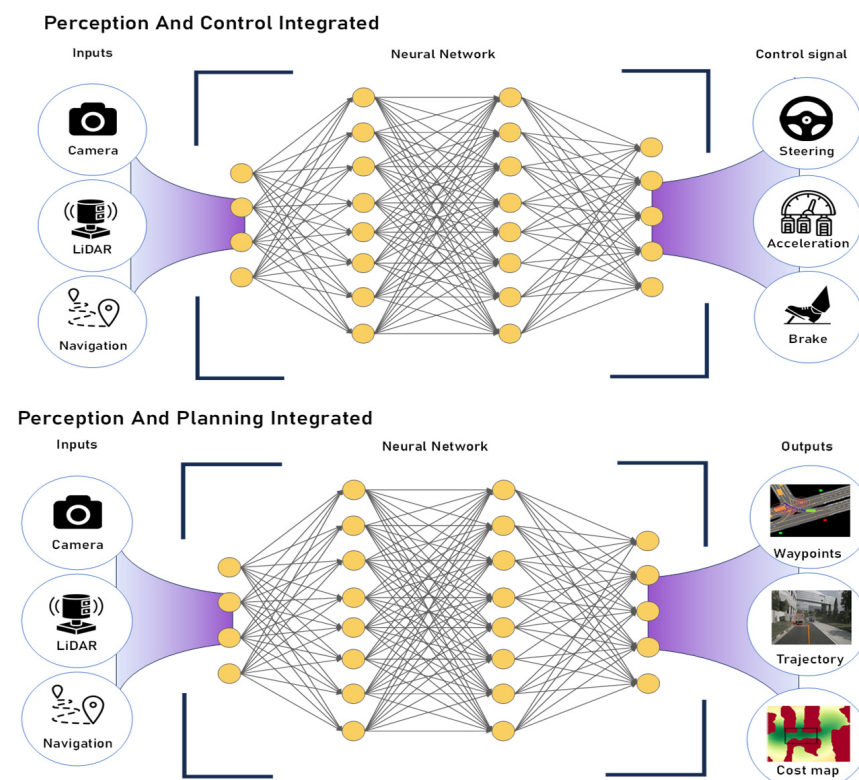


Figure 2. Perception–control integration and perception–planning integration. In the top figure, neural policy maps multimodal observations directly to low-level commands (steering, acceleration, brake). In the bottom figure, the network outputs planning representations such as Waypoints/trajectory/cost map, which are consumed by a downstream tracking controller to produce low-level commands. The distinction is based on the output layer type, the existence of intermediate planning representations, and the coupling relationship with the downstream control module.

- Perception and Control Integrated

This approach represents a single network that directly maps raw sensor inputs to low-level control signals such as steering angle or acceleration, without explicit trajectory or semantic output. Typical works include refs. [3,5,29,30]. Paniego et al. [29] integrated the convolutional long short-term memory (ConvLSTM) memory variant based on PilotNet [3] and introduced vehicle speed as an additional input channel, so that a single controller has short-term dynamic perception capabilities. This is an end-to-end control system trained through imitation learning, where the network directly outputs low-level commands rather than Waypoints or trajectories. However, the disadvantage is that the training is prone to instability when the memory window is too long, and there is a lack of interpretable intermediate trajectories.

Perception and control integration represents the simplest path to end-to-end autonomous driving: no intermediate semantics are required, and latency is minimal. With the introduction of visual memory, kinematics knowledge, BEV, and other content, the robustness of this paradigm across diverse conditions has been continuously improved.

- Perception and Planning Integration

This method indicates that, during the inference phase, the neural network itself generates an intermediate planning output, typically in the form of Waypoints or a cost map. These outputs are then sent to a feedback controller, which continuously tracks and predicts the trajectory. Unlike perception–control integration, where the network emits low-level controls directly, this paradigm uses intermediate planning outputs that are sent to a feedback controller for trajectory tracking and control. Some works (refs. [31–35]) include intermediate planning outputs. Hu et al. [31] proposed a vision-based end-to-end framework that combines self-centered alignment accumulation techniques with dual-path modeling methods. This approach enhances the interpretability of spatiotemporal feature fusion and intermediate representations. The inference output is a planning representation used by downstream tracking controllers rather than directly by low-level commands. The framework achieved success in driving success rates and collision counts within the CARLA closed-loop simulation. SparseDrive [33] generates planning trajectories by replacing dense BEV tensors with sparse instance tokens. Under the nuScenes open-loop planning protocol reported in [33], SparseDrive-B reduced the average L2 error from 0.72 m to 0.58 m and the collision rate from 0.21% to 0.06% compared with the UniAD baseline. MomAD [34] stabilizes trajectory planning outputs through momentum-consistency attention and reports improved closed-loop driving performance on Bench2Drive. Because nuScenes and Bench2Drive use different evaluation regimes, these results are described separately rather than directly compared.

This paradigm features end-to-end global optimization and can be enhanced at the planning level through techniques such as sparse representation and temporal constraints, thereby improving vehicle performance in both open-loop and closed-loop scenarios. Moreover, this paradigm is evolving toward unified Transformer-based architectures [36]. Unlike traditional end-to-end CNN-based models that rely on feature flattening, these architectures leverage Transformer to maintain spatially structured representations throughout the backbone network. This design choice simultaneously addresses scalability requirements, with performance expected to improve as both data scale and model capacity increase.

Despite its architectural advantages, optimizing this integrated model remains challenging. The primary obstacle is gradient conflicts [37] arising from multi-task learning. A phased approach, first freezing the perception task and then fine-tuning the planning task, has proven to be a viable solution for resolving these conflicts. Furthermore, while the Transformer backbone supports long-term sequence reasoning, efficiently implementing

it requires addressing substantial computational overhead. Further technical discussions regarding specific Transformer variants and their implementation details will be covered in Section 3.4.

2.3. Input Modality

- **Camera:** The camera is one of the most commonly used input modalities, capturing texture, color, and semantic information of the scene. In modular architecture, camera data is explicitly decomposed using a modular approach, and intermediate results are used for subsequent modules. In end-to-end architecture, camera data are directly used as input, with features being automatically extracted using deep learning. For instance, refs. [31,38] employed a multi-camera system to construct a BEV, the former leveraging the captured data to generate dynamic predictions for the future and the latter enhancing the image features. Similarly, refs. [4,8,39,40] utilize cameras as a primary input, underscoring their significance in providing visual information. Ref. [41] implements semantic segmentation using camera inputs to develop a road scene segmentation model suitable for autonomous driving. However, camera performance can be significantly affected under challenging lighting conditions, such as intense sunlight, rain, or snow [42,43]. These limitations highlight the need for robust preprocessing or fusion with other sensor modalities to enhance reliability in real-world scenarios.
- **LiDAR:** LiDAR is a key input modality in end-to-end autonomous driving due to its ability to generate precise 3D point clouds, which provide detailed spatial geometry and distance information. Unlike cameras, LiDAR is less affected by environmental factors, such as low light, glare, or partial occlusion. This makes it particularly effective for detecting obstacles and dynamic targets with high accuracy. For instance, ref. [44] used LiDAR data for multimodal trajectory prediction by jointly modeling the interactions between multiple traffic participants. Ref. [45] trained vehicle driving strategies in complex urban interaction scenarios through direct inputs to LiDAR point clouds, and both articles have the integration of LiDAR data for perception and planning in an end-to-end system in common. Furthermore, ref. [46] explored self-supervised learning to utilize LiDAR data under conditions of limited labeled data, while ref. [47] achieved higher accuracy by transforming LiDAR point clouds into BEV RGB mappings, outperforming traditional LiDAR-only methods. Despite its advantages, the high cost of LiDAR systems remains a major barrier to their widespread adoption, especially for consumer-level autonomous vehicles.
- **Navigational Inputs:** The navigation inputs are mainly used to specify the vehicle's travel goals and tasks, and in the end-to-end model, these inputs are directly used as part of the network. For example, the target location is provided by GPS, which transforms the navigation inputs into a set of sparse path points [8]. Additionally, highly accurate environmental information, such as lane lines, is also provided, and training signals are constructed via high-definition maps (HD maps). In this case, the model uses modular privileged information to learn the environmental features in the training phase but ultimately generates the control signals through the end-to-end network [48]. Ref. [35] employs HD maps to provide a BEV of the environment, which enhances the model's understanding of the road layout and dynamic targets, and ultimately generates trajectories close to real driving in the simulation environment. Ref. [7] provides the current position of the vehicle via GPS and utilizes HD maps to provide road semantic information, which is ultimately transformed into a BEV in a unified manner, enabling the end-to-end model to handle multiple driving tasks simultaneously. When navigation input is used as the input modality, it is usually

combined with BEV [49,50], which is a very common research direction to improve end-to-end autonomous driving performance. Finally, navigation inputs may be affected by signal interference in high-density building areas, and the cost of real-time updating and maintenance of HD maps is relatively high.

2.4. Output Modality

- **Direct Control Signals:** Direct control signals are arguably the most common output modality and are particularly suitable for scenarios requiring high real-time control and low complexity. Direct control signals map input data to driving behavior through an end-to-end network, avoiding intermediate modules in traditional architectures. Typically, these signals include steering angle and acceleration. Refs. [3,31,51–53] all use steering angle as an output since it directly determines vehicle orientation. Refs. [54,55] utilize acceleration as the output of longitudinal control, given that longitudinal control dictates the vehicle's forward speed or braking effort, thereby ensuring that the vehicle maintains its target speed or a safe distance from the vehicle in front. While lateral control governs the steering angle and longitudinal control dictates the acceleration, a more detailed analysis of the specific application is necessary to fully comprehend the intricacies of these systems. It is important to note that direct control signals are subject to limitations in dynamic and complex scenarios, particularly in terms of their global optimization capacity. Consequently, direct control signals are frequently employed in conjunction with other output modes.
- **Waypoints:** Waypoints differ from direct output control signals in that path points are a collection of target points that the vehicle needs to pass through in a future period, usually as part of path planning. It is noted that although the goal of an end-to-end system is to map directly from inputs to outputs, this does not mean that end-to-end architectures cannot include mid-level representations. In some designs, Waypoints are explicitly generated to reduce learning complexity. Ref. [35] proposes an end-to-end autonomous driving model that incorporates mid-level representations to directly generate future Waypoints, which are translated into actual vehicle control commands via low-level control modules. The Waypoints generated in [56] are a set of future trajectories describing the desired position of the vehicle at the planning time, using a cost function to optimize the selection of the Waypoints. The model in [57] integrates features from RGB cameras and LiDAR, fuses these features at multiple resolutions through a multi-layer Transformer module, and finally generates Waypoints, which in the end directly guide the vehicle's low-level control module (PID controller) to generate steering, acceleration, and braking signals. The challenge about the target Waypoint is that its prediction delay may affect the system's performance.
- **Multi-Task Outputs:** This refers to the simultaneous prediction of multiple tasks in the same model, which improves the overall system performance and enables the co-optimization of the end-to-end system. It is worth mentioning that, in direct control, signals such as steering angle and acceleration are generally present as output modes at the same time, which is the output mode in most of the papers [54,56,58]. In the domain of path planning, there are also Waypoints and future trajectories together as outputs, resulting in a substantial system. Ref. [59] proposed a lightweight multimodal fusion model with outputs including Waypoint predictions for future trajectories and semantic segmentation maps based on RGB-D data. The main outputs of ref. [60] are the prediction of BEVs of future driving scenarios and the generation of dynamically constrained smoothed trajectories based on the predicted trajectories, where a large-scale self-supervised pre-trained Transformer model is used, which is well adapted for dynamic scenarios. Multi-task joint output is demanding in terms of weight allocation

among tasks, and conflicting objectives may lead to performance degradation, while its demand on computational resources limits the feasibility of large-scale deployment.

3. Key Techniques and Paradigms

3.1. Imitation Learning (IL)

3.1.1. Behavioral Cloning (BC)

BC treats imitation learning as supervised learning: a policy is trained on expert state–action pairs to directly mimic expert behavior [61]. The model learns a mapping from the current state to the corresponding action. We categorize driving tasks by complexity, considering one-lane or highway driving and simple lane changes as simple and intersections or urban driving as complex. Early works [3,28,62,63] used BC to achieve single-lane keeping by learning the mapping from state s to action. Ref. [61] indicates that BC can coordinate both lateral and longitudinal control in previously unseen settings, enabling certain complex tasks. However, behavioral cloning is vulnerable to covariate shift: the state distribution in expert demonstrations differs from that encountered at test time. A small initial prediction error can push the agent into an out-of-distribution state s' . Lacking experience on how to recover from s' , the agent tends to take another suboptimal action, causing errors to compound and ultimately leading to policy failure. The solution to this problem in the domain of end-to-end autonomous driving is contingent upon the implementation of interactive learning methodologies, such as Dataset Aggregation (Dagger) [64], which empower agents to systematically acquire new states encountered during execution. Subsequently, experts designate the appropriate actions for these states and incorporate them into the training data. This process is repeated to continuously refine the policy, thereby compensating for the limitations of the training data and enhancing the robustness of the system in real-world environments. The efficacy of this approach has been demonstrated in [65,66].

3.1.2. Inverse Reinforcement Learning (IRL)

IRL is a specific form of learning by demonstration that attempts to estimate the reward function of a Markov Decision Process based on examples provided by an expert. And the reward function is often considered to be the most concise description of the task [67]. In ref. [68], IRL was used to model drivers' lane-changing behavior, reproducing individual trajectories that were then fed into a low-level motion controller to perform the underlying lane change operations. In refs. [69–71], maximum entropy IRL is used as the core method for modeling human driving behavior. This approach addresses the ambiguity problem of identifying a reward function by selecting a function that can explain expert behavior while minimizing uncertainty in other behaviors. This is typically represented by associating the probability of a trajectory with its reward, and the situations considered are all complex conditions, such as multi-lane lane changes or intersections, with the ultimate goal of trajectory planning and reward function optimization to ensure that the system behavior is as close as possible to expert performance. Compared to BC, IRL is more suitable for handling complex scenarios because it can better generalize through the derivation of reward functions. With the continuous improvement of IRL technology, Generative Adversarial Imitation Learning (GAIL) [72–74] has been proposed. GAIL is composed of two components: a generator, which is the agent's policy π_θ that endeavors to generate state–action pairs that are indistinguishable from those of the expert policy, and a discriminator that, following training, can differentiate between state–action pairs in the expert demonstration dataset and those generated by policy π_θ . Both the discriminator and the generator are trained in an adversarial loop, with the discriminator's output serving as a dynamic reward signal learned by the generator. The generator (policy) is updated

using policy gradient algorithms to maximize this reward, effectively encouraging it to generate behavior that deceives the discriminator. This process directly optimizes the policy to match the expert's data distribution without explicitly defining a reward function. Given its capacity to circumvent the constraints imposed by conventional IRL, thereby facilitating more precise and reliable policy learning, GAIL is regarded as a relatively recent approach within the context of end-to-end autonomous driving.

Imitation learning enables the rapid learning of effective driving strategies by directly utilizing expert data or inferring potential reward functions. However, imitation learning is also limited by the bias problem and high dependence on expert data. In addition, insufficient coverage of expert behaviors in complex tasks may also lead to limited policy generalization capabilities.

3.2. Reinforcement Learning (RL)

RL [75,76] is a machine learning approach that enables an agent to interact with the environment and learn from feedback. The objective of RL is to derive a policy (π) that guides the agent in selecting actions in each state to maximize the expected cumulative reward. In the context of autonomous driving, this translates to learning driving strategies that allow vehicles to operate safely and efficiently across different scenarios. Unlike imitation learning, RL does not rely on expert demonstrations and can adapt its policy through interaction, thereby addressing IL's limitations in complex, dynamic settings. Table 2 provides a detailed comparison of the studies leveraging IL and RL approaches, highlighting the specific technologies used.

In end-to-end autonomous driving, the general classification of reinforcement learning approaches includes value-based approaches, which learn a value function over states or state–action pairs; policy-based approaches, which optimize a parameterized policy directly; and fusion approaches, which combine value estimation with policy optimization. Below, we outline the characteristics of each class and their typical use in end-to-end driving.

3.2.1. Value-Based Approach

The value-based approach focuses on estimating future returns. Specifically, it uses the state–action value function $Q(s, a)$ to measure the expected cumulative reward under a given policy π . In the subsequent action selection, the action with the highest Q value is implicitly generated. The advantage of this approach is that it does not require explicit storage or updating of the policy network.

For end-to-end autonomous driving, the objective is to maximize the expected long-term return under safety and efficiency considerations. Some common methods include the Deep Q-Network (DQN) method [77,78], which is an extension of Q-Learning and is used to handle multi-dimensional state spaces. However, it is worth noting that, when it comes to value-based approaches, there are also drawbacks, such as insufficient safety constraints (further discussion on safety will be found in Section 4). Therefore, Ahmed et al. [77] concatenated RGB images with dynamic features such as vehicle speed and heading angle, and input them into DQN to improve the policy stability. Haritz et al. [78] introduced safety shielding based on DQN, filtering unsafe actions through post-safety constraints and combining intermediate features such as scene geometry and right of way to improve the interpretability and generalization ability of Q -value estimation.

Table 2. Comparison of imitation and reinforcement learning methods in end-to-end autonomous driving.

Method	Reference	Specific Technology	Input	Output
Imitation Learning (IL)	Chen et al. [62]	Behavior Cloning	State–action pairs	Steering & Brake
	Li et al. [63]	Behavior Cloning	Camera image	Steering
	Liu et al. [69]	Inverse Reinforcement Learning	Driving trajectories	Lane Change
	Wu et al. [70]	Inverse Reinforcement Learning	State–action pairs	Control Actions
	Ho et al. [72]	Generative Adversarial Imitation Learning	State–action pairs	Control Actions
	Li et al. [73]	Generative Adversarial Imitation Learning	Driving trajectories	Steering & Brake
Reinforcement Learning (RL)	Peng et al. [79]	Double Deep Q-Network	Camera image	Steering & Brake
	Cai et al. [80]	Deep Q-Learning	Vehicle state	Speed & Lane
	Anzalone et al. [81]	Proximal Policy Optimization	Camera image	Steering & Brake
	Zhao et al. [82]	Proximal Policy Optimization	Camera image	Steering & Brake
	Siboo et al. [83]	Deep Deterministic Policy Gradient	Vehicle state	Steering & Brake
	Zou et al. [84]	Deep Deterministic Policy Gradient	Vehicle state	Steering & Brake
	Song et al. [85]	Smooth Actor–Critic	Vehicle state	Steering & Brake

After that, in order to reduce Q-value overestimation and accelerate learning, the value-based approach now typically replaces traditional DQN with its variants, such as Double-DQN [86] for curbing overestimation bias and Dueling-DQN [87] for decoupling state value and advantage for more efficient learning.

3.2.2. Policy-Based Approach

Compared to value-based methods, this approach directly maximizes the expected cumulative return obtained from the parameterized policy $\pi_{\theta}(a|s)$ without explicitly estimating Q or V. π_{θ} is a probability mapping composed of state s and action a , representing the conditional probability distribution of taking action a in state s . Representative methods include the classic REINFORCE algorithm [88]. However, in the field of end-to-end autonomous driving, due to the high variance and low sample efficiency of the pure policy method training process, it is rarely applied directly in practice. Compared with the aforementioned value-based approach, the pure policy approach has the advantages of direct decision-making and ease of implementation. However, under the continuous-control requirements of actual autonomous driving tasks, its convergence stability and efficiency are significantly insufficient, so it is generally not used as an actual solution. To address these issues, mainstream research in end-to-end autonomous driving reinforcement learning tends to adopt the Actor–Critic (AC) architecture, which integrates policy optimization with value function estimation.

3.2.3. Fusion Approach

Value-based methods such as DQN are not suitable for continuous steering because simple discretization introduces quantization and instability. Conversely, pure policy gradient methods often have high variance and low sample efficiency. The Actor–Critic framework decouples policy execution (Actor) from value assessment (Critic), combining the sample efficiency of value-based methods with the continuous-control advantages of policy-based methods. As shown in Figure 3, the Actor (policy) selects an action based on the current state. It learns a policy that maps states to actions. The Critic (value function)

estimates the value of a state or state–action pairs [89]. In addition, the decoupling of Actor and Critic allows the model to improve scalability. In what follows, we group all methods with an Actor–Critic structure under this class.

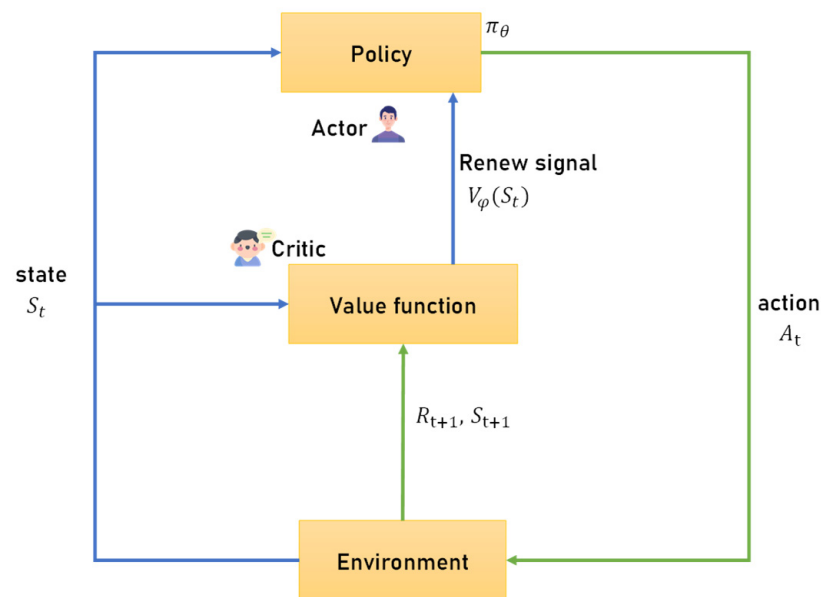


Figure 3. Actor–Critic interaction loop. At time t , the state S_t is fed into both the policy π_θ (Actor) and the value function (Critic). The policy outputs an action A_t that acts on the environment, which returns a reward and the next state: R_{t+1}, S_{t+1} . The Critic calculates $V_\phi(S_t)$ to update π_θ .

Among Actor–Critic methods, PPO [81,82,90], DDPG [83,84,91], and SAC [92] differ mainly in how they collect and reuse data, parameterize continuous-control policies, stabilize policy updates, and balance exploration with safety. These differences lead to distinct stability–efficiency trade-offs when they are applied to high-dimensional, safety-critical driving tasks.

PPO is an on-policy stochastic Actor–Critic method designed for stable policy updates by limiting excessive policy changes. This stabilizing effect is useful for high-dimensional vision-based driving tasks. Uppuluri et al. [93] used a PPO-Clip variant to avoid divergence caused by excessive policy step sizes. Under the CARLA Town07 closed-loop testing protocol reported in [93], the method achieved a success rate of 92%. This result should be interpreted as a within-protocol result rather than a direct comparison with results obtained on other CARLA towns or benchmarks.

DDPG targets continuous control through an off-policy deterministic Actor–Critic framework, making it suitable for steering, throttle, and braking decisions. In end-to-end driving, DDPG is attractive because it naturally handles continuous actions, but it can be sensitive to hyperparameters, reward design, and exploration noise. Rizehvandi et al. [94] modeled the decision layer as a continuous-action Markov Decision Process (MDP) and reported a 31% reduction in dangerous lane change rate relative to DQN and PPO, indicating the advantage of deterministic continuous control for fine-grained decision-making.

SAC combines off-policy learning with entropy-regularized exploration, improving sample reuse and robustness in continuous-control tasks. However, entropy maximization can induce unsafe over-exploration in safety-critical driving scenarios. Chu et al. [95] integrate barrier functions into SAC to enforce safety constraints while retaining exploration benefits, though failures may still increase at higher speeds, highlighting the need for speed-aware safety shaping or shielding. Different from standard Soft Actor–Critic, Song et al. [85] proposed a Smooth Actor–Critic method for end-to-end autonomous driving,

which introduces an action-smoothness regularization term to constrain the difference between neighboring steering, acceleration, and braking actions.

The Actor–Critic architecture decouples policy optimization from value estimation, combining variance reduction from value learning with the continuous-control capability of policy optimization, and thus serves as a widely used formulation in end-to-end driving research. Nevertheless, AC by itself offers no inherent safety guarantees; in safety-critical driving tasks, it is commonly paired with explicit safety mechanisms to curb unsafe exploration and improve constraint satisfaction, while achieving formal system-level guarantees remains an open challenge.

Current evidence for reinforcement learning-based end-to-end driving remains concentrated in simulations rather than real-world deployment. PPO-based methods have shown closed-loop competence in specific CARLA settings, whereas DDPG- and SAC-based methods are mainly valued for continuous-control learning and exploration efficiency. However, their demonstrated performance remains bounded by simulator fidelity, scenario coverage, reward design, and the availability of explicit safety constraints. Therefore, RL formulations are promising for interactive closed-loop learning, but they still require safety shielding, constraint design, and broader closed-loop validation before production-scale deployment.

3.3. *Transfer from Simulator to Real*

To avoid the high costs of using real cars, researchers often train end-to-end autonomous driving models in simulation, but the goal is still to apply the model to real cars. However, due to the differences between the simulation environment and the real world in terms of lighting, vehicle dynamics, etc., directly deploying the model to reality will encounter a reality gap [96]. To this end, domain adaptation methods have emerged to improve the cross-domain generalization ability of end-to-end autonomous driving models. That is, the task remains the same, but a domain shift or distribution change occurs between the source and target distributions [97]. A common technique for domain adaptation is self-supervised learning [98–100], which can be trained on unlabeled data by designing predefined tasks that let the model learn more robust intermediate representations. Despite reducing the need for manual labeling, a large amount of real-world data is still required for pre-training, so this remains a major challenge for this approach. This issue is avoided by cross-domain adaptation through an unsupervised approach [101,102], which involves fine-tuning a model trained in simulation using a large amount of unlabeled real data. Zhao et al. [103] proposed ePointDA, an end-to-end simulation-to-real domain adaptation framework for LiDAR point cloud segmentation. The method bridges the pixel-level domain gap by rendering dropout noise for synthetic LiDAR data and reduces the feature-level gap through statistics-invariant and spatially adaptive feature alignment, thereby improving transfer from synthetic GTA-LiDAR to real KITTI and SemanticKITTI. Hu et al. [104] propose a UDA framework for lane detection that jointly optimizes an adversarial generator and discriminator to narrow the domain gap. Traditional CycleGAN-based sim-to-real methods mainly adjust image style, which does not guarantee suitability for end-to-end policies. To solve this problem, Rao et al. [105] proposed RL-CycleGAN, which introduces the RL scene consistency loss that will make the simulated images closer to reality.

Sim-to-real transfer via domain adaptation reduces labeled real-data requirements, alleviating the data bottleneck for IL and RL. However, the current method still cannot eliminate the distribution gap between simulation and reality, which may lead to performance degradation of the migrated model.

3.4. Transformer

Transformers leverage the self-attention mechanism to model dependencies among inputs, effectively capturing long-range context in sequences and multimodal data.

In end-to-end autonomous driving, Transformers are used to process multimodal fusion and extract BEV features [57,106], owing to their unified Query–Key–Value (QKV) interface, which allows all modalities to be embedded into the same dimensional vector space and passed to the self-attention layer. They can also serve as cross-modal scene encoders: self-attention enables interactions among BEV, map, and agent tokens, thereby encoding lane geometry and temporal offsets [107] and even generatively enforcing collision free constraints [108]. In this context, GenAD [108] uses a Transformer to explicitly impose vehicle dynamics constraints to avoid physically infeasible trajectories, which is also the key to surpassing traditional neural networks.

Another advantage of the Transformer is its scalability; this means that performance improves predictably as the model and data scale increase. Compared with Recurrent Neural Networks (RNNs), self-attention provides shorter information-flow paths and imposes weaker inductive biases, enabling larger models to better exploit multimodal cues. Recent work [109] illustrated that, while keeping the dataset unchanged, when the model parameters are increased from 47 M to 646 M, the Driving Score improves by 23 points, and the collision rate and trajectory error decrease simultaneously. This result indicates a scaling trend within that specific evaluation setting rather than a directly comparable gain across different benchmarks. Nevertheless, this scalability comes at the cost of the quadratic computational complexity of the self-attention mechanism, which leads to a sharp increase in inference latency and computational requirements, posing a significant challenge for real-time vehicle deployment. To mitigate these challenges, researchers are exploring several strategies. One direct approach is to introduce sparsity into the architecture [33,34], which reduces the number of calculations required in the attention layers. Another prominent trend is the use of diffusion-based planning models. These models not only perform well in generating coherent long-term trajectories but also address the planning challenges mentioned in Section 2.2. For example, some diffusion-based planners can achieve real-time performance at 45 FPS [110], while others can predict complex multimodal trajectories for over 10 s [111], showcasing their potential to balance high performance with computational tractability. In addition to computational cost, large-scale Transformer and end-to-end models also depend on sufficient data scale, data quality, and long-tail coverage, which are further discussed in Section 8.1.

Transformer-based end-to-end architectures can be summarized according to the main bottlenecks they address. BEV-centric spatiotemporal Transformers enhance cross-view and temporal fusion for structured scene representation [57,106]. Efficiency-oriented variants, such as SparseDrive [33] and MomAD [34], use sparse queries or temporal consistency mechanisms to reduce attention overhead and stabilize planning outputs. Constraint-aware generative designs introduce vehicle dynamics or collision-free constraints, while diffusion-based planners combine Transformer backbones with generative trajectory modeling [110,111]. Under the nuScenes open-loop planning protocol with ST-P3 metrics, DiffusionDrive [110] achieved an average L2 error of 0.57 m, compared with 0.72 m for VAD and 0.61 m for SparseDrive under the reported ResNet-50 setting. These variants show how Transformer-based models balance representation quality, inference efficiency, physical feasibility, and trajectory diversity in end-to-end driving.

3.5. World Model

World models have recently seen broad adoption in end-to-end autonomous driving. They can synthesize diverse, realistic driving scene rollouts, enriching training data and

enabling the simulation of long-tail events (such as pedestrians suddenly crossing the road), as shown in Figure 4. We also summarize recent driving world models in Table 3, organizing them by time. And Guan et al. [112] generated high-resolution, statistically consistent driving scenes through a world model to enrich complex interactive scenes. With richer data and scenarios, systems not only recognize surrounding agents more reliably but also better anticipate their intentions, demonstrating the predictive utility of world modeling. The typical work GAIA-1 [113] captures spatiotemporal dynamics and high-level structures in driving scenes through a diffusion decoder, modeling the world as an unsupervised sequence modeling problem, and then predicts the next token in the sequence. DriveDreamer [114] extends the conditional diffusion framework, jointly predicting future frames and actions in the form of yaw angle and velocity.

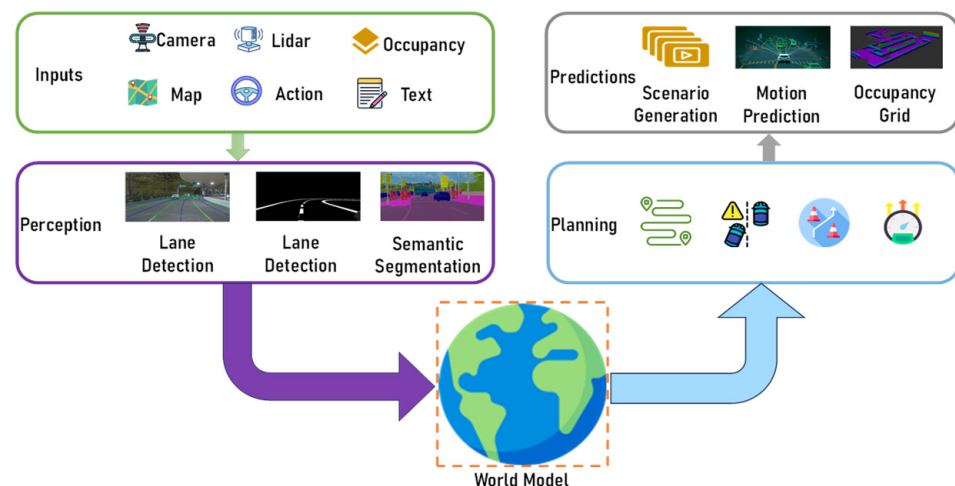


Figure 4. End-to-end autonomous driving framework based on world model.

World models are particularly effective at improving generalization in long-tail scenarios. DriveDreamer-2 [115] explicitly utilizes large language model (LLM)-guided semantic reasoning to enhance robustness in long-tail scenarios, while DriveWorld [116] and UniWorld [117] both achieve effective multi-scenario generalization through a unified latent representation.

However, there are still limitations in the control of actions by world models. Although some works [115,118,119] have already linked world models to driving videos, fine-grained control of actions remains extremely challenging. Secondly, actions are continuous variables, making it difficult to learn their latent spatial representations from limited data samples. Current methods work better for nominal motion control and exhibit gaps under aggressive maneuvers (e.g., sustained sharp turns). Another limitation is related to data. Effective action inference and prediction require large, diverse datasets, yet current collections largely rely on vehicle-mounted passive sensors without active sensing or attention, which limits coverage and diversity. Furthermore, the long-tail problem remains an unsolved issue, which is a critical component for autonomous driving. Although Guan et al. [112] released a dataset containing many real-world crash scenarios, such efforts remain insufficient. Addressing these limitations is essential for advancing world model research in end-to-end autonomous driving.

Table 3. Overview of representative driving world models from 2023 to 2024.

World Model	Timeline	Core Mechanism	Key Innovation	Evaluation Focus
Uniworld [117]	September 2023	Unified pre-training	Pre-training on multimodal data in a shared latent space	On nuScenes, outperforms monocular pre-training by +1.5 IoU in motion prediction, +2.0 mAP/+2.0 NDS in 3D detection, and +3.0 mIoU in scene completion; 3D annotation cost is reduced by 25%.
DriveDreamer [114]	September 2023	Conditional diffusion	Generating driving videos conditioned on structured signals for long-horizon scenarios	On nuScenes under the ST-P3 open-loop planning setting, achieves 0.29 m avg. L2 and 0.15% collision; 4K synthetic images improve BEVFusion from 32.8 to 35.8 mAP.
GAIA-1 [113]	September 2023	Autoregressive Transformer world model	Industry-scale unsupervised sequence modeling with large parameters	Evaluated mainly as a generative driving world model, using 25 Hz to 6.25 Hz video subsampling for longer-horizon rollout.
MUVO [120]	November 2023	Multimodal generative	Spatial-centric multimodal world modeling: predicts multiple future modalities in a unified latent rollout	Evaluates multimodal future-scene prediction against BEV-based baselines, using PSNR for image prediction and Chamfer distance for point cloud prediction.
Drive-WM [121]	November 2023	World model-assisted planning eval	Consistent multi-view video generation with a unified condition interface and factorized generation for multi-view temporal consistency	On nuScenes video generation, improves on DriveDreamer from 52.6/452.0 to 15.8/122.7 in FID/FVD.
Drive-Occworld [122]	December 2023	Vision-centric 4D occupancy	Forecasting world model; vision-centric 4D occupancy and flow forecasting with semantic- and motion-conditional normalization in memory	On nuScenes-Occupancy, evaluates 4D occupancy forecasting; its planning variant reports 0.47 m avg. L2 and 0.11% avg. collision in open-loop planning.
DriveDreamer-2 [115]	March 2024	LLM-enhanced diffusion	LLM-enhanced semantic reasoning for controllable long-video generation	For customized multi-view video generation, improves over Drive-WM and DrivingDiffusion, reaching 11.2 FID and 55.7 FVD.

4. Safety

Ensuring safety in end-to-end autonomous driving systems is essential for safe vehicle operation and the protection of other road users, and it underpins public trust and acceptance of autonomous driving technologies.

As a highly promising approach, end-to-end systems necessitate ensuring safety across the entire pipeline. Due to their high dependence on deep neural networks, these systems are susceptible to adversarial attacks. Adding stickers to traffic signs [123] or using clothing patterns to deceive computer vision systems [124] have been proven to easily cause the system to make incorrect judgments. In addition, Knox et al. [125] pointed out that almost all existing RL work on autonomous driving has the risk of reward mismatch, which directly increases the probability of collision and violation, destroys offline indicators, and makes the verification process “blind”.

These attack vectors necessitate developing evaluation methods that quantify their real-world impact. Zhang et al. [126] also point out that combining models with driving behavior analysis is necessary to comprehensively assess the actual safety of autonomous driving systems. Because current evaluations are largely limited to closed-loop simulation or formal verification, a practical first step is a static reward audit to align learned objectives with safety goals [125]. Following this, Zhu et al. [127] propose a novel simulation testing method based on intelligent vehicle-to-vehicle interaction, which uses adversarial yet safe intelligence to simulate dangerous driving behaviors in the real world to create safety assessment scenarios that more closely resemble real-world situations. Lars et al. [128] combine conformal prediction with formal verification to validate the safety of autonomous systems. This method can identify potential safety risks during the static reward audit phase, thereby ensuring system security. However, overly conservative predictions may affect system performance.

Although these assessment methods can expose problems, they cannot reduce risk by themselves. Therefore, we must introduce additional safety mechanisms into the decision-making and control pipeline to proactively mitigate the above threats. Some common methods include adding safety constraints [129–131], safety-optimized learning methods [132,133], and adding safety trigger mechanisms [134,135]. These three groups differ in their safety assumptions and residual risks. Constraint-based methods rely on explicit risk models and real-time optimization to restrict unsafe trajectories. Safety-optimized learning methods improve policy robustness through constraints, adversarial objectives, or uncertainty-aware learning, but their effectiveness remains affected by reward design and scenario coverage. Safety trigger mechanisms provide an additional intervention layer when unsafe behaviors are detected, but they may introduce conservative responses or over-intervention in complex scenarios. Among them, adding safety constraints is the most common method, including safety cost functions, obstacle avoidance strategies, and unsafe factors. Gharavi et al. [130] directly included collision probability and behavioral risk as costs, while imposing a chance constraint $\epsilon = 10^{-3}$ on the entire predicted trajectory to ensure that the global collision risk was controlled. Their Max–Min–Plus–Scaling Mixed-Integer Linear Programming (MMPS-MILP) formulation converted the nonlinear probability function into a piecewise affine form and obtained globally optimal solutions in 96% of the reported time steps with a 0.2 s sampling interval. This result reflects the optimization performance of the proposed constrained planner rather than a general driving-success metric. He et al. [132] introduced two-level Lagrange penalties λ_1 and λ_2 to model environmental uncertainty as an adversarial agent, enabling the vehicle to learn to successfully merge even under worst-case conditions, with an average success rate of 96% across three traffic density levels in the simulated merging experiments reported in [132].

However, these successes remain insufficient to cover unknown long-tail scenarios, revealing the limits of current safety techniques. Some works [136,137] explore combining world models with formal safety guarantees. Li et al. [136] predict future latent features while using safety rules to filter out actions that violate speed limits in the potential space. In the CARLA Town05 experiment reported in [136], filtering the latent action space yielded a collision rate of 6% while maintaining task success rates comparable to the unshielded strategy. Meanwhile, safety in end-to-end settings increasingly entails ethical and societal considerations. Misjudgments can trigger over-intervention, introducing new hazards. Building public trust and complying with laws, regulations, and standards are prerequisites for the large-scale deployment of end-to-end autonomous driving. We discuss these issues in Section 7.

5. Interpretability

Interpretability [138,139] refers to making the decision-making process of an AI system transparent and understandable so that researchers and users can understand the model's reasoning. In autonomous driving, interpretability is particularly important because it determines the predictability, debuggability, and trustworthiness of the system's behavior. It is particularly challenging for end-to-end architectures compared with traditional modular systems. In modular pipelines, each component has well-defined inputs, outputs, and processing steps, allowing researchers to analyze intermediate results and reconstruct the decision logic stage by stage. However, the end-to-end architectures use deep neural networks to map sensor inputs directly to control outputs, lacking explicit intermediate representations. Due to the lack of explicit module boundaries, it becomes challenging to explain the system behavior and trace the rationale behind specific decisions [140].

Interpretability is not merely an academic concern; in real-world deployments, it is a core determinant of the deployability and optimization of end-to-end autonomous driving systems. Due to the lack of explicit middle layers and decomposable decision paths in end-to-end architectures, the ensuing problem is model generalizability and behavior consistency [14]. Advancing interpretability not only facilitates model understanding and optimization but also strengthens public acceptance. Accordingly, to understand the interpretability of an end-to-end autonomous driving system, we need to first analyze the basis of each decision, that is, how the model makes judgments under specific inputs [141]. Beyond describing what a method explains, interpretability also requires discussing how the quality of an explanation can be assessed. Commonly discussed criteria include faithfulness and stability [142], while autonomous driving explanations also need to consider human-oriented plausibility and acceptability [138–140]. Faithfulness concerns whether an explanation reflects the actual decision mechanism of the model; stability evaluates whether similar inputs, adjacent frames, or repeated runs produce consistent explanations; and plausibility assesses whether the explanation is understandable and acceptable to human users. These criteria are important for end-to-end autonomous driving because a visually reasonable explanation may not necessarily reflect the causal factors used by the policy network.

5.1. Local Reasoning

In an end-to-end autonomous driving system, local reasoning refers to the instantaneous decision logic of the driving policy network under specific input conditions, that is, why the system makes the current control decision at a certain moment and in a certain environment. This helps us to understand the local interpretability of the system. Typical aspects include (i) the model's focus regions and (ii) decision consistency. The core focus area is often the saliency map [143–145] and the attention mechanism [8,106,146,147].

Their purpose is to know what features the model pays attention to in the input. Porres et al. [10] employed pre-computed saliency semantic maps to constrain the Transformer's self-attention, incorporating a loss term during training to encourage the model to prioritize safety-critical elements such as vehicles and pedestrians. This "explanation as supervision" paradigm [148] can be viewed as an alignment folding of the feature manifold: it shrinks the policy search space and provides inherent robustness to adversarial noise. This represents a shift whereby, rather than being mere observers after training, explanations become "rule-makers" during the training phase.

However, focusing solely on regional restrictions is insufficient to assess policy robustness; it is also necessary to examine decision consistency. Decision consistency is typically probed via counterfactual or ablation-based input interventions [149,150] because such interventions often affect some of its features, and the main purpose is to understand the model's decision logic. Wu et al. [149] tested the causal decision logic of the large language model by changing the semantic input in four ways. Feng et al. [151] intervened in driving environment variables during the training process, including removing non-critical states and reconnecting critical states, thereby improving the learning efficiency of the neural network and making it pay more attention to safety-critical events.

When the system fails to distinguish decisions arising from normal inputs versus intervention-induced inputs, behavior tracing becomes unreliable and epistemic uncertainty increases. Nevertheless, testing decision consistency through input intervention is a necessary method. First, observing changes in control outputs allows one to trace causal chains. Second, extending interventions across multiple dimensions, such as temporal and hidden layers, enables local reasoning to access policy robustness, thereby enhancing model interpretability further. Such counterfactual and ablation-based interventions are also useful for evaluating faithfulness because they test whether the factors identified by an explanation actually influence the model output.

In recent years, self-explaining models have attracted growing interest for local reasoning in end-to-end autonomous driving. Self-explaining networks offer alternative internal paths for local reasoning, which are entirely distinct from the saliency map and input intervention. The structure of the self-explanatory model as shown in Figure 5. These networks embed explanatory mechanisms within their structure, enabling training and explanation to be performed simultaneously. Alvarez-Melis et al. [142] implement concept-based explanations via prototypes and encourage diversity through sparsity regularization, using a staged training procedure to obtain simple local explanations. SENN reports strong quantitative scores across standard explanatory criteria.

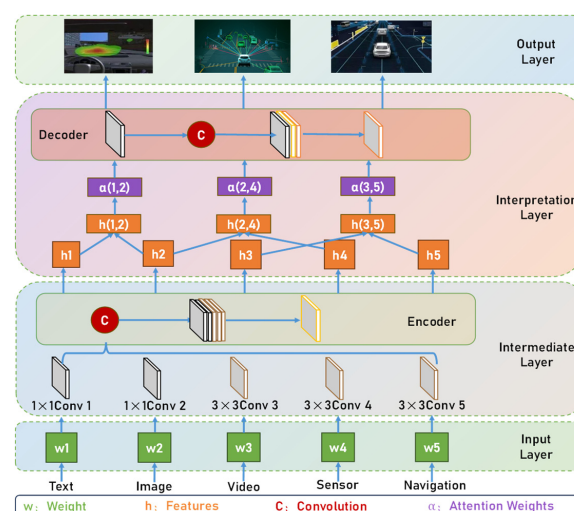


Figure 5. Architecture of a self-explaining model.

However, all local reasoning methods essentially only interpret the control logic of a “single moment” and are unable to reveal the global behavior patterns of the model in long time series and multiple scenarios. To thoroughly assess the explainability and safety boundaries of end-to-end systems, it is necessary to shift from a local perspective to a global level of explanation.

5.2. Global Reasoning

In an end-to-end autonomous driving system, one must assess a model’s long-horizon behavior and consistency in complex scenarios. Local reasoning can reveal why this action is being taken, but it is difficult to answer what the vehicle’s future actions will be and to what extent, so explainability must be elevated to the long-term, scene-level dimension. By characterizing global behavior over extended trajectories, we improve model transparency and interpretability, motivating a shift from local to global reasoning.

Long-term driving strategy is the core part of global reasoning, which represents the vehicle’s ability to globally plan and dynamically adjust future driving behavior. Two prominent lines of work are generative prediction [152,153] and causal modeling [154,155]. Zhao et al. [152] pointed out that, by generating possible future trajectories, the system can predict and intervene in complex driving scenarios, improving cross-scenario consistency. Xu et al. [153] proposed predicting the long-term control strategy of the vehicle by combining video input and language models, while providing natural language explanations of the corresponding behaviors. This method enhances both long-term behavior prediction and the transparency of decisions in realistic driving settings.

However, generative predictors do not explain the causal mechanisms behind their decisions. Because long-horizon planning depends on a multi-step chain of causality, causal reasoning can be used to assess policy reliability and improve interpretability. To compensate for the shortcomings of generative methods, which only provide results without revealing the decision-making chain, Hu et al. [154] proposed a causal modeling framework that predicts intentions and generates future trajectories, thereby strengthening long-term planning. This causal modeling approach has also been successfully applied in other works [156,157].

Although global reasoning makes up for the shortcomings of local reasoning that only targets a single decision, global reasoning usually requires modeling a large number of historical trajectories, environmental variables, and interactive behaviors, which significantly increases the computational cost. Moreover, because the policy must reason over longer horizons and scene-level contexts, system latency increases, and responsiveness can degrade. Therefore, global reasoning should also be examined in terms of explanation stability across temporal horizons and scene-level plausibility, rather than only whether it can generate an explanation for a single decision.

6. Benchmarking

6.1. Datasets

In end-to-end autonomous driving, rich and diverse-scenario datasets underpin model generalization and robustness, ensuring stable performance in complex real-world environments. However, different datasets are applied to different scenarios: for example, KITTI and A2D2 cover foundational driving scenes; BDD100K and DriveLM target diverse, large-scale settings; and ApolloScape and nuScenes provide a large amount of urban scenarios data, focusing on fine-grained labeling. The details are provided in Table 4.

Table 4. Summary of end-to-end autonomous driving datasets.

Datasets	Year	Modalities							Content				Size	Location	
		Cam.	LiDAR	GNSS	Steer.	Spd./Acc.	Nav.	Route	Traffic	Lighting	Weather	Time			Obj.
KITTI [158]	2013	✓	✓	✓					✓		✓		✓	43 k	Karlsruhe
BDD 100 K [159]	2018	✓		✓					✓		✓		✓	1100 h	US
ApolloScape [160]	2018	✓	✓						✓		✓		✓	147 k frames	China
Waymo [161]	2019	✓	✓	✓	✓	✓					✓	✓	✓	5.5 h	California
nuScenes [162]	2019	✓	✓	✓					✓		✓	✓	✓	5.5 h	Boston, Singapore
A2D2 [163]	2020	✓	✓	✓	✓	✓			✓		✓		✓	390 k frames	South of Germany
nuPlan [164]	2022	✓	✓	✓		✓		✓	✓				✓	1200 h	Las Vegas, Boston, Pittsburgh, Singapore
Argoverse V2 [165]	2023	✓	✓	✓		✓			✓	✓	✓	✓	✓	1000 h	US
DriveLM [166]	2023	✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	4.5 h	Boston, Singapore
ZOD [167]	2023	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓	100 k	European
DriveCoT [168]	2024	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	5 h	CARLA
MARS [169]	2024	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	500 k frames	US
CoVLA [170]	2024	✓		✓	✓	✓			✓	✓	✓	✓	✓	80 h	Tokyo, Japan

“Cam.” denotes camera, “Steer.” denotes steering, “Spd./Acc.” denotes speed or acceleration, “Nav.” denotes navigation, and “Obj.” denotes objects. Annotation types and licensing/access conditions vary across datasets and should be checked from the official dataset sources before use.

The annotation process of end-to-end autonomous driving always contains multi-dimensional information, and everyone's annotation style is different. To improve consistency at scale, weakly supervised labeling is often adopted to reduce manual effort, but it may introduce label noise; rigorous quality control is therefore required. Based on this, future dataset design should focus first on extreme scenarios and long-tail problem coverage; second on high-quality annotation and multi-tasking support; and third on cross-scenario and cross-region to adapt to the road traffic rules of different cities.

6.2. Evaluation Methodology

Evaluation methodologies are central to ensuring the performance and safety of end-to-end autonomous driving systems. Current practice is commonly categorized into open-loop and closed-loop evaluation, each with distinct strengths and limitations. This section systematically analyzes their technical principles, applicable scenarios, and constraints, and also reviews widely used simulation platforms to support rigorous evaluation and optimization of end-to-end systems.

6.2.1. Open-Loop Evaluation

Open-loop evaluation is a key method for assessing the performance of end-to-end autonomous driving models. It involves using predefined datasets as input to the model, which generates control commands or outputs without interacting with the environment or receiving feedback. This evaluation focuses on the model's ability to make accurate decisions based on input data and evaluates the stability and reliability of its decision-making logic in an ideal, static environment.

Common metrics include error, accuracy, and collision rate. The primary advantage of open-loop evaluation is its efficiency. It does not require complex real-world testing environments or large-scale field deployments. Instead, the process can rely on pre-collected datasets, which significantly reduces testing costs and resource requirements. Many studies use the nuScenes dataset to evaluate performance, relying on metrics such as the L2 error between predicted and actual trajectories and collision rates [171]. However, these metrics have limitations. The L2 distance and collision rate alone do not fully capture the quality of the model's planning behavior [172]. Moreover, the lack of interaction and feedback with the environment creates a gap between the model's performance in virtual conditions and real-world scenarios. Closed-loop evaluation complements open-loop testing by placing the model in an interactive environment (simulation or real-world), thereby providing a more faithful assessment of real-world behavior while addressing the limitations of purely offline evaluation.

6.2.2. Closed-Loop Evaluation

Closed-loop evaluation is a critical component of end-to-end autonomous driving research, as it directly applies model decisions to dynamically changing vehicle and environmental conditions. Compared with open-loop testing, it better reflects real driving. In this approach, control outputs at time t affect the vehicle's state and subsequent sensor inputs at $t + 1$, forming a complete perception–decision–control feedback loop. Common metrics for closed-loop evaluation include the Driving Score [173] and the Route Completion Rate [5]. One key advantage of closed-loop evaluation is its ability to assess the long-term stability and robustness of models in complex scenarios.

A persistent challenge is the sim-to-real gap: simulators may not fully capture real-world conditions, so results can over- or underestimate performance. Mitigations include high-fidelity vehicle dynamics for accurate motion responses and multi-agent simulations with dynamic traffic participants to reflect realistic flows and pedestrian behavior. While the

sim-to-real gap cannot be eliminated, steadily improving simulator realism and standards can narrow it and yield results that better predict on-road performance.

Closed and open-loop evaluations are complementary. Open-loop testing enables rapid iteration and screening on curated datasets; promising models are then validated in a closed loop under dynamic, interactive conditions to ensure robustness before deployment.

6.2.3. Simulation Platforms

For the current evaluation criteria of end-to-end autonomous driving performance, most of them are based on simulation platforms, and only a few of them are based on realistic simulation scenarios or real vehicle tests. These data show that simulators remain the de facto tool because they provide safe, controllable, and reproducible environments for extensive experimentation—including corner-case scenarios—without exposing real vehicles or pedestrians to risk, and they enable rapid iteration at lower cost. We have also summarized the current simulation platforms, including whether the simulator is open source to use, etc. Detailed information is shown in Table 5.

Table 5. Functional and benchmark comparison of simulation platforms for end-to-end autonomous driving systems.

Simulators	Open Source	Engine Base	Environment & Weather	Key Indicators	Typical End-to-End Task	Scene Library	Primary Application Scenarios
CARLA	✓	Unreal	Dynamic Day–Night Cycle	Collision Rate Route Completion Late-Break Penalty Infractions	ST-P3 [31] MomAD [34] TransFuser [57] LMDrive [174]	Town01-10	Complex urban driving, long-tail corner cases, end-to-end benchmarking
AirSim	✓	Unreal	Photorealistic Rendering	Lane-Center Deviation Velocity RMS Error Collision Rate	Zhao et al. [175] Aslam et al. [176]	Unreal Blocks Custom Assets	High-fidelity visual simulation
LGSVL	✓	Unity	Unity HDRP & Sensor Effects	Parking Success Path Deviation Time-to-Park	Lu et al. [177]	Unity Cities, Unity Highways, Unity Parking Lots	Vehicle-to-everything (V2X) testing
DRIVE Sim		Ominiverse	RTX Ray-Tracing & Physics-Based	RSS Violation Ratio Sensor Latency	Not E2E-specific	Omniverse USD Scene Store	Large-scale fleet sim
PreScan		Proprietary	Physics-Based Sensor Noise	AEB Activation ACC Spacing Error Steering Oscillation	ADAS validation, not E2E-specific	MATLAB Scene Generator	ADAS validation
MetaDrive	✓	Panda3D	Parametric Weather & Light	Success Rate Collision Rate Off-Road Rate Safety Violations	Generalizable RL driving policy learning	Procedural Road-Block Maps	Generalization evaluation, generation of safety-critical scenarios

Among the various simulation platforms, CARLA [173] stands out as a leading choice due to its ability to provide realistic urban driving environments and simulate diverse traffic conditions, including vehicles, pedestrians, and weather variations. These capabilities are critical for the training and testing of models. CARLA supports a wide range of sensors, offering robust testing and validation opportunities for the perception module. Its comprehensive sensor integration and adaptability make it suitable for a variety of applications, from academic research to industrial deployment. This versatility has established CARLA as one of the most prominent simulation platforms in end-to-end autonomous driving research.

7. Applications of End-to-End Architecture

7.1. Tesla FSD

Tesla’s Full Self-Driving (FSD) is a camera-centric technology stack. Tesla employs a vision-based occupancy network as its core scene representation and does not rely on high-definition maps. The architecture is shown in Figure 6. Planning is performed in BEV space with an occupancy-based network, which helps mitigate reliance on high-definition maps, simplifying deployment [135].

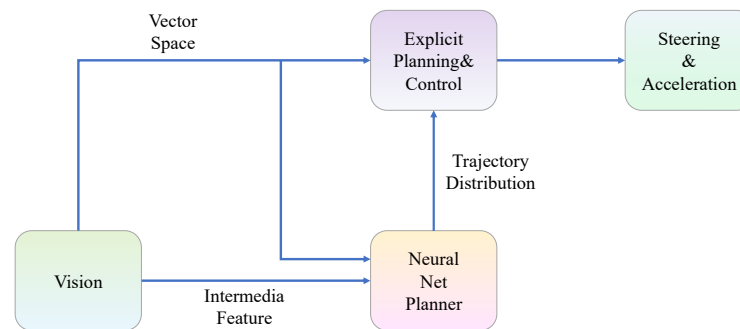


Figure 6. Tesla’s end-to-end architecture. The vision input is processed by a neural network planner, while explicit planning and control modules convert the learned planning output into steering and acceleration commands.

From an industrial perspective, Tesla’s closed-loop learning can be viewed as a “fleet-data engine-training-fleet” pipeline featuring selective data collection and scalable annotation capabilities. First, FSD employs an onboard trigger mechanism to determine whether to capture and transmit training clips, rather than uploading raw data streams, thereby supporting fault-driven or novelty-driven sampling under bandwidth constraints. Second, Tesla has publicly discussed a clip-centric data pipeline in which multi-camera “clips” uploaded from vehicles are processed by multiple offline neural networks to generate intermediate products, followed by optimization steps that produce vector space supervision for training. This clip-centric pipeline is conducive to throughput and scalability in large-fleet closed-loop iteration because it shifts the bottleneck from transmission volume to data curation and label-generation throughput.

For sim-to-real validation and safety regression, Tesla has publicly discussed simulation as a complementary capability rather than a replacement for fleet data. Specifically, simulation is used to support scalable data generation and to cover scenarios that are difficult, rare, or unsafe to collect consistently from real-world driving. Tesla also emphasizes photorealistic camera-level simulation for producing training supervision in its data pipeline [178]. Additionally, Tesla’s “shadow mode” runs candidate models silently in parallel with the deployed stack and human driving to log discrepancies, serving as an industrially practical form of online regression gating and data mining [178].

In summary, Tesla’s occupancy-based scene representation and pure vision approach can reduce reliance on maps. Their feasibility in the real world hinges on closed-loop learning at the fleet scale—specifically, selective fragment mining, scalable automated annotation, and release gating mechanisms. This coupling between representation choices and data engine capabilities represents the closed-loop perspective we must consider in the autonomous driving industry.

7.2. Xpeng XNGP

Xpeng XNGP (Navigation Guided Pilot) provides a unique industry reference for the computational power and engineering efficiency required to deploy end-to-end architectures. Its technology stack includes XNet (perception), XPlanner (learning-based

planning), and XBrain (cognition), representing a paradigm shift toward occupancy-centric representation combined with learning control strategies.

XNet employs a static–dynamic integrated network to output real-time 2K-resolution occupancy grids, explicitly reconstructing navigable space and fine-grained obstacles with over 2 million grid cells. This dense representation serves as unified input for XPlanner, enabling the system to transition from rule-based trajectory generation to a neural planner that evolves through large-scale data ingestion, aiming to achieve human-like driving characteristics.

This shift to an occupancy-based end-to-end paradigm has shifted the primary industrial bottleneck from model design to pipeline throughput. For example, Xpeng reported an automated annotation workflow for approximately 500,000 video clips, completing the process within 16.7 days [179]. This figure describes data pipeline throughput rather than model accuracy, but it illustrates the infrastructure requirements of industrial E2E deployment. This demonstrates that the feasibility of end-to-end autonomy in industrial applications is not constrained by algorithmic capabilities, but rather by the ability to build high-fidelity automated annotation pipelines. Additionally, deployable scalability is a hard constraint. An unoptimized single network may consume the budget for a single NVIDIA Orin-X GPU, as opposed to engineering optimization. This demonstrates that system-level compression is a prerequisite for achieving large-scale end-to-end model deployment in vehicles. This further underscores that end-to-end scalability in production environments hinges on infrastructure throughput and iteration frequency.

Consequently, progress in industrial end-to-end models should be viewed as a closed-loop process constrained by data, annotation, training, and deployment scalability—not merely an isolated architectural choice.

7.3. Horizon Robotics SuperDrive

Horizon Robotics’ strategy does not merely view end-to-end autonomous driving as an algorithmic choice, but rather as a problem of coupling representation with infrastructure. Only when software is co-designed with the underlying computational infrastructure can a unified end-to-end architecture be deployed at scale.

At the perception level, SuperDrive demonstrates a fusion-based end-to-end perception architecture that integrates dynamic perception, static perception, and occupancy estimation into a single pipeline. This architecture is designed to address production bottlenecks such as high latency, excessive rules, and heavy workloads. Concurrently, Horizon positions SuperDrive as a Transformer-based end-to-end solution and emphasizes software–hardware co-optimization for production deployment. The architecture is shown in Figure 7.

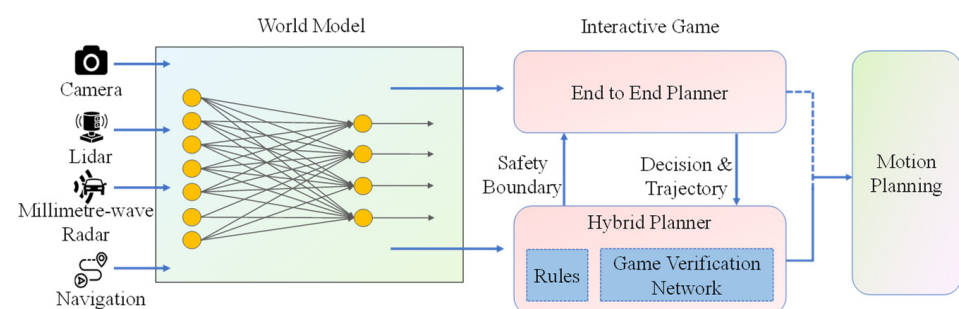


Figure 7. Horizon Robotics’ end-to-end architecture. The diagram illustrates the coupling between an end-to-end planner and a hybrid planner, where rule constraints and game-verification modules are incorporated before generating the final motion planning output.

However, the feasibility of this technology is constrained by deployment scalability. Horizon addresses this by coupling its algorithm stack with the Journey 6 platform and the Nash BPU architecture [180]. In particular, its public technical description states that Nash BPU is engineered to support state-of-the-art Transformer networks and provides an automotive-oriented toolchain to meet low-latency edge constraints. Thus, Horizon's case suggests that end-to-end progress is not merely an isolated software upgrade: the practical benefits of unified occupancy-centric perception and data-driven interaction depend heavily on infrastructure efficiency and toolchain maturity on edge devices, not just network architecture.

7.4. Baidu AIR ApolloFM

AIR ApolloFM is an open source end-to-end driving system designed for real-world vehicle deployment. The architecture is shown in Figure 8. At the model architecture level, ApolloFM introduces a Vision–Language–Action (VLA) foundation model as its starting point, then refines it into a deployable model rather than first training a small policy. Its output is explicitly designed as a triplet: open-set perception signals, scene descriptions, and direct driving actions. This output design supervises “understanding and intent” beyond mere geometry while preserving the action interface for end-to-end control. ApolloFM subsequently refined this VLA into a real-time 4D full-scene perception model and an end-to-end driving model. The 4D perception model is described as a refined VL model with controlled computational capabilities for real-time execution, outputting an open-set world model containing dense representations, instance-level representations, and inter-element topological relationships. For the driving model, ApolloFM defines three deployment modes to balance inference capabilities and latency budgets.

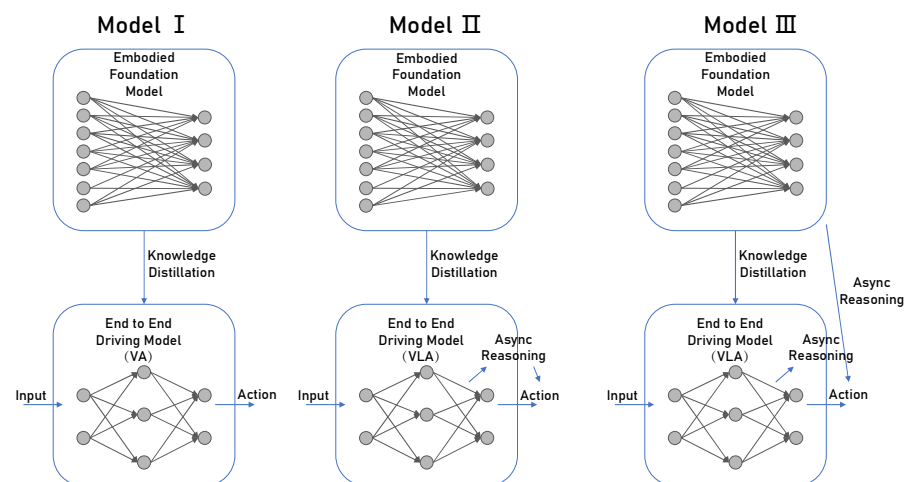


Figure 8. Baidu AIR Apollo FM's end-to-end architecture. The architecture refines an embodied foundation model into VLA-based end-to-end driving models, supporting perception, scene reasoning, and driving action generation under different deployment modes.

Overall, ApolloFM highlights a deployable end-to-end driving foundation model pathway. It begins with a VLA model featuring perception–description–action supervision, then refines it into real-time 4D world modeling and driving strategies with explicit latency–capability trade-offs.

8. Challenges and Future Trends

8.1. Challenges

8.1.1. Data and Long-Tail Problem

Significant progress has been made in recent years in the construction of end-to-end autonomous driving datasets. However, data-related challenges continue to constrain their performance. These challenges fall into two aspects: firstly, the architecture requires a large-scale data supply. As we mentioned in Section 3.4, the widespread adoption of complex architectures such as Transformer has led to significant improvements in the ability of end-to-end systems to capture temporal correlations and multimodal information. However, all of this relies on the availability of large volumes of diverse, precisely labeled data. In reality, an insufficient data scale makes it difficult for models to learn rich representations, which is one of the key challenges we face. A common consequence of this is underfitting or overfitting. Additionally, multimodal sample coverage is incomplete. For instance, the nuScenes dataset contains data from six cameras, but only a single LiDAR sensor. This further weakens the model's ability to fuse information from various environments. Therefore, despite having strong model design capabilities, performance bottlenecks will still be caused by data size.

The second issue is the long-tail problem and the imbalance in data distribution. The long-tail problem refers to extremely rare traffic scenarios that pose a significant risk when they occur, such as vehicles driving in the wrong direction or sudden obstacles. This is because the occurrence of extreme events is extremely unlikely, making data collection difficult. If they are created artificially, safety and legality issues will become obstacles. Additionally, accurately labeling long-tail scenarios is challenging, which is why current datasets tend to have a wealth of conventional scenarios but a shortage of extreme ones. Moreover, the majority of existing datasets are limited to specific cities and countries, for example, the difference in driving directions between the UK and China, resulting in a narrow distribution range. This means that most datasets are domain-specific and lack geographic diversity, which directly leads to poor model generalization. In other words, the model performs well in carefully designed benchmark tests but struggles in dangerous situations that have not been encountered in the real world.

Currently, simulators and world models show great potential for enhancing long-tail scenario data and reducing annotation costs. Some works [181,182] have already achieved success in this area. However, sim-to-real and cross-domain adaptation remain difficult due to mismatches in real-world physics, sensor characteristics, and behavioral diversity, creating generalization bottlenecks.

8.1.2. Human-like Decision-Making

An excellent autonomous driving system must not only comply with traffic rules, which is the most basic requirement, but also anticipate other agents' intentions and respond intuitively in a human-like manner. It must be closer to human drivers in terms of logic and behavior and have intuitive judgment and cognitive processes similar to humans. Current end-to-end models still have a significant gap in incorporating human-like decision-making. This is because models lack explicit cognitive mechanisms and normative priors encoding ethics and social norms, which makes human-like judgment difficult in safety-critical scenarios. Incorporating cognitive science principles into algorithm design is very complex, and some works [183,184] suggest that autonomous driving will shift towards behaviors closer to those of humans. Despite these efforts, human-like decision-making remains a significant challenge to achieving fully autonomous driving systems that can predict more vehicle behaviors and react intuitively.

8.1.3. Limited Interpretability

Although Section 5 provides a detailed discussion of interpretability, limited interpretability in practice remains a major challenge for end-to-end autonomous driving. Most current end-to-end models do not expose transparent decision rationales, which leads to: (i) reduced public trust, (ii) constrained ability for developers and auditors to evaluate and diagnose model decisions, and (iii) weak post-incident root-cause analysis and accountability in real-world deployments.

Although current interpretability technologies, such as attention mechanisms and global reasoning, have solved some interpretability problems, further exploration is still needed, including multi-scale explanations [185] and further construction of a set of interpretability evaluation rules for systems and humans to understand.

These challenges are not isolated but interdependent. Data scarcity and long-tail distributions motivate the use of cross-scenario international datasets, as well as high-fidelity digital twins, to enhance generalization. In parallel, the inadequacy of human-like decision-making has led to a gap in cognitive mechanisms, which has driven research towards more explainable and human-like decision-making. The limits of interpretability often reflect weak semantic and spatial understanding, underscoring a shift from passive perception to active 3D spatial reasoning. Therefore, future research not only needs to continue to explore each single issue in depth, but also needs to coordinate and comprehensively consider the interactions between these challenges as a whole.

8.2. Future Trends

8.2.1. Scene Transition and Internationalization

For end-to-end systems to be applied efficiently, they must overcome the bottleneck of excellent training in one domain but ineffective training in others. The Road–Waymo dataset [186], which is based on the Waymo dataset and integrates US and UK traffic scenarios, has demonstrated the feasibility of cross-city joint training. Therefore, future research into creating cross-scenario and cross-city joint datasets will provide a more robust data foundation for end-to-end autonomous driving models.

8.2.2. Cognitive Mechanism

In order to move towards end-to-end decision-making combined with human-like decision-making, based on the reference to cognitive psychology and neuroscience [187], a driver's cognitive decision-making relies on the simple mental representation or topological abstraction of the brain's reconstruction of the driving scene; that is, the cognitive decision space should contain the interactive relationship of key information of the driving scene [188]. Some works, such as [183], constructed a "Wide Neural Network (WNN)" model to simulate the complex neural network of the human brain to achieve personalized cognition and human-like driving intention decision-making. Lu et al. [184] combined cognitive mapping with hierarchical reinforcement learning to construct a three-layer structure of "cognition–decision–control," thereby achieving more human-like lane change timing selection. In order to make this cognition operational and credible, it can be combined with normative priors and ethical constraints so that the strategy can handle inevitable collision dilemmas or value trade-offs in a socially acceptable way [189]. Specifically, some ethically relevant indicators, such as accident rates and trust indexes, are incorporated into the evaluation phase to monitor public acceptance [190]. Algorithms that incorporate cognitive mechanisms can make the output easy to plan and at the same time contain interpretable principles based on the cognitive layer, enabling it to predict and negotiate more like humans, improving predictability, trust, and safety under the condition of end-to-end trainability. Therefore, future end-to-end policies can build scene-graph representations

on an explicit cognition layer and pair these with normative priors encoding social and ethical constraints.

8.2.3. Digital Twin

Although there are already many excellent simulation environments for simulating the real world in the field of end-to-end autonomous driving, most of them still have the shortcomings of insufficient fidelity of physical engines and simulation engines, and poor dynamic interaction. A digital twin can establish a “digital mapping body” that is completely consistent with the physical object or system, using a high-fidelity simulation environment to test the boundaries and weaknesses of the end-to-end system to ensure real-road safety. Simultaneously, this high-fidelity simulation environment enables systematic regression testing by tracking model behavior under safety-critical conditions through the replay and reconstruction of real-world accident scenarios, thereby providing robust support for long-tail scenarios.

Recent progress indicates that digital twin technology is increasingly being integrated with Sim2Real migration to further bridge the gap between simulation and reality. On one hand, high-fidelity, sensor-calibrated digital twins can generate effective synthetic training data. Shahbaz et al. [191] trained LiDAR-based perception models with digital twin data using the same architecture and reported an AP of 44.74, compared with 42.70 when using real data. This within-study comparison suggests that high-fidelity, perception-aware twins can reduce the domain gap rather than merely augment data. Adapters based on foundation models provide a complementary route by aligning appearance and distribution statistics. Samak et al. [192] reported that this strategy bridged more than 40% of the perceptual Sim2Real gap and reduced the average style difference by 40.33%.

Moreover, data on long-tail safety-critical events in the real world remain extremely scarce. Therefore, a promising research direction is to combine cross-domain adaptation with world model-based generation and apply it to digital twins. Specifically, the world model can serve as a scenario engine to synthesize rare interaction events, thereby extending the long-tail coverage for training and regression testing. Meanwhile, cross-domain adaptation functions as a distribution corrector, aligning the generated twin data with the target domain to avoid introducing new domain shifts. Therefore, future research holds promise for developing a closed-loop process that leverages real-world data records for twin calibration and cross-domain adaptation, integrates world model-driven long-tail scenario generation, and enables large-scale closed-loop evaluation to achieve iterative refinement under long-tail conditions.

8.2.4. Spatial Understanding

For end-to-end autonomous driving, accurate and intuitive spatial understanding is imperative. The model provides critical spatial and semantic priors for planning, as well as laying the foundation for robust decision-making under various environmental conditions. The capacity to effectively combine and interpret sensor data to generate a comprehensive, three-dimensional understanding of the driving environment is what the spatial understanding brings. NuSceneQA [193] is a visual question answering (VQA) benchmark designed for autonomous driving scenarios proposed by Qian et al. Although it focuses on the positional relationships of single objects and ignores spatial relationships between objects, it is still a pioneering work on spatial understanding. SURDS [194] fill this gap by introducing absolute spatial reasoning and relative position reasoning tasks and proposing a set of evaluations to assess the spatial understanding capabilities of multimodal large language models in autonomous driving scenarios. Therefore, using a progressive chain of reasoning will enhance the model’s spatial understanding capabilities, upgrading

passive perception to reasoning-driven active decision-making, which will in turn promote further learning in end-to-end autonomous driving scenarios.

8.2.5. Multi-Agent Collaborative End-to-End Driving

Despite significant progress in end-to-end autonomous driving within single-vehicle scenarios, decision-making still relies primarily on limited perceptual information from self-agents and partially observable interaction data. This limitation is particularly pronounced in long-tail safety-critical scenarios such as occlusions, complex lane changes, and intersection navigation, where the agent's perception alone cannot reconstruct the complete multi-agent state. Consequently, multi-agent cooperative end-to-end driving is emerging as a critical development trend.

Some works indicate that V2X collaboration can enhance the robustness of end-to-end planning. UniV2X [195] evaluates this idea using two cooperative planning benchmarks: DAIR-V2X, a real-world vehicle–infrastructure cooperative dataset, and V2X-Sim, a CARLA-based simulated V2X dataset. In the DAIR-V2X planning evaluation, UniV2X reduced the average collision rate from 0.89% to 0.49% relative to the no-fusion baseline and improved L2 planning errors at 2.5 s/3.5 s/4.5 s from 1.62/2.64/3.97 m to 1.31/2.05/3.06 m. In the V2X-Sim evaluation, vehicle-to-vehicle cooperation reduced the average collision rate from 2.00% to 1.75% and improved the corresponding L2 errors from 1.47/2.61/4.02 m to 1.33/2.43/3.68 m. Notably, UniV2X also shows that collaborative supervision can enhance intermediate scene understanding, suggesting that V2X collaboration may improve both perception robustness and downstream planning safety.

Beyond accuracy, end-to-end cooperative driving must also address practical constraints such as bandwidth, latency, and message redundancy. A prominent research direction is communication-driven multi-agent learning, where agents learn when and what to communicate. IntNet [196] proposes a communication-driven multi-agent reinforcement learning framework and demonstrates that selective interaction simultaneously improves task success rates and efficiency. This highlights a core future trend: multi-agent end-to-end systems should eliminate communication redundancy—moving beyond inefficient continuous transmission to build a semantically aware adaptive strategy that enables intent-driven, efficient interconnection. Meanwhile, robust collaborative perception remains a prerequisite for cooperative decision-making, particularly in the presence of attitude errors and asynchrony issues—two persistent challenges in real-world V2X deployments. V2X-ViT2 [197] enhances Transformer-based V2X collaborative perception technology and demonstrates superior robustness in noisy environments.

Overall, multi-agent collaborative end-to-end driving represents a future development trend, as it directly addresses the fundamental limitation of observability constrained solely by self-perception and offers a practical pathway to enhance long-tail coverage and safety. Future research may explore enabling V2X collaboration under scalable coordination mechanisms, ultimately delivering reliable safety benefits under real-world long-tail conditions. The five directions discussed above are at different stages of development. Scene transition and internationalization are supported by increasingly diverse datasets and closed-loop benchmarks, but robust cross-region generalization remains unresolved. Digital twins and spatial understanding are rapidly emerging directions for long-tail testing, simulation-based validation, and reasoning-aware perception, although their effectiveness still depends on simulation fidelity, annotation quality, and evaluation standards. Cognitive mechanisms and multi-agent collaborative end-to-end driving may provide broader long-term benefits, but they still face challenges in human-like reasoning, social norm modeling, communication latency, and scalable coordination.

9. Conclusions

This review systematically analyzes this cutting-edge field and proposes a functionally oriented analytical framework that divides existing architectures into perception and control integration and perception and planning integration. We delve into key paradigms such as imitation learning and reinforcement learning and illuminate the impact of transformative technologies such as the Transformer and world models. We also discussed safety and interpretability, which are two important attributes for end-to-end autonomous driving. We summarized industrial deployment cases in order to link technology with practical applications. In real vehicles, end-to-end architectures must eventually interact with low-level execution systems, including steer-by-wire, braking actuators, throttle control, and longitudinal control hardware. Therefore, actuator dynamics, control latency, command smoothness, and tracking reliability remain important factors when translating planned trajectories or direct control outputs into physical vehicle motion. Our analysis highlights three interdependent challenges—data and long-tail coverage, human-like decision-making, and limited interpretability—which in turn motivate five research directions: cross-scenario and international generalization, cognitive mechanism, high-fidelity digital twin evaluation, spatial understanding, and multi-agent collaborative end-to-end driving. These will continue to be hot topics and key issues in the end-to-end autonomous driving field. For production deployment, the key bottleneck is not only model accuracy, but also the closed-loop engineering pipeline that connects data collection, long-tail scenario mining, model updating, evaluation, and vehicle execution. Therefore, future end-to-end autonomous driving systems should be evaluated not only by planning accuracy, but also by whether their outputs can be executed smoothly, timely, and reliably by downstream actuation and control systems.

Author Contributions: Conceptualization, Y.C. and G.Y.; methodology, Y.C. and G.Y.; formal analysis, Y.C. and G.Y.; investigation, Y.C. and G.Y.; writing—original draft preparation, Y.C. and G.Y.; writing—review and editing, P.R. and Z.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Hubei Province Major Science and Technology Innovation Program (2024BAA011), in part by Hubei Province key Research and Development Project (2025BAB082), in part by the Hubei provincial Natural Science Foundation Joint Fund for Innovation and Development Project (2024AFD045) and in part by the Guided Project of Scientific Research Plan of Hubei Provincial Department of Education for (2024B2024436).

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BC	Behavioral Cloning
BEV	Bird's-Eye View
CNN	Convolutional Neural Network
Dagger	Dataset Aggregation
DDPG	Deep Deterministic Policy Gradient
DQN	Deep Q-Network
FSD	Full Self-Driving
GAIL	Generative Adversarial Imitation Learning
HD Map	High-Definition Map
IL	Imitation Learning

IRL	Inverse Reinforcement Learning
LLM	Large Language Model
MDP	Markov Decision Process
MILP	Mixed-Integer Linear Programming
MMPS	Max–Min–Plus–Scaling
MPC	Model Predictive Control
PID	Proportional–Integral–Derivative
PPO	Proximal Policy Optimization
QKV	Query–Key–Value
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SAC	Soft Actor–Critic
V2X	Vehicle-to-Everything
VLA	Vision–Language–Action
VQA	Visual Question Answering

References

1. Singh, S. *Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey*; Traffic Safety Facts—Crash Stats, Art. no. DOT HS 812 115; National Center for Statistics and Analysis: Washington, DC, USA, 2015. Available online: <https://trid.trb.org/View/1346216> (accessed on 16 March 2025).
2. Tampuu, A.; Semikin, M.; Muhammad, N.; Fishman, D.; Matisen, T. A Survey of End-to-End Driving: Architectures and Training Methods. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 1364–1384. [[CrossRef](#)] [[PubMed](#)]
3. Bojarski, M.; Del Testa, D.; Dworakowski, D.; Firner, B.; Flepp, B.; Goyal, P.; Jackel, L.D.; Monfort, M.; Muller, U.; Zhang, J.; et al. End to End Learning for Self-Driving Cars. *arXiv* **2016**, arXiv:1604.07316.
4. Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. Planning-oriented Autonomous Driving. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 17853–17862.
5. Codevilla, F.; Muller, M.; Lopez, A.; Koltun, V.; Dosovitskiy, A. End-to-End Driving via Conditional Imitation Learning. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 4693–4700.
6. Xu, H.; Gao, Y.; Yu, F.; Darrell, T. End-to-End Learning of Driving Models from Large-Scale Video Datasets. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2174–2182.
7. Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D.; Han, S. BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird’s-Eye View Representation. *arXiv* **2022**, arXiv:2205.13542.
8. Prakash, A.; Chitta, K.; Geiger, A. Multi-Modal Fusion Transformer for End-to-End Autonomous Driving. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 7073–7083.
9. Chen, Z.; Ye, M.; Xu, S.; Cao, T.; Chen, Q. PPAD: Iterative Interactions of Prediction and Planning for End-to-end Autonomous Driving. *arXiv* **2024**, arXiv:2311.08100.
10. Porres, D.; Xiao, Y.; Villalonga, G.; Levy, A.; López, A.M. Guiding Attention in End-to-End Driving Models. In Proceedings of the 2024 IEEE Intelligent Vehicles Symposium (IV), Jeju Island, Republic of Korea, 2–5 June 2024; pp. 2353–2360.
11. Zhang, Q.; Tang, M.; Geng, R.; Chen, F.; Xin, R.; Wang, L. MMFN: Multi-Modal-Fusion-Net for End-to-End Driving. *arXiv* **2022**, arXiv:2207.00186.
12. Huang, Z.; Sun, S.; Zhao, J.; Mao, L. Multi-modal policy fusion for end-to-end autonomous driving. *Inf. Fusion* **2023**, *98*, 101834. [[CrossRef](#)]
13. Ishihara, K.; Kanervisto, A.; Miura, J.; Hautamaki, V. Multi-task Learning with Attention for End-to-end Autonomous Driving. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 19–25 June 2021; pp. 2896–2905.
14. Chen, L.; Wu, P.; Chitta, K.; Jaeger, B.; Geiger, A.; Li, H. End-to-end Autonomous Driving: Challenges and Frontiers. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 10164–10183. [[CrossRef](#)] [[PubMed](#)]
15. Chib, P.S.; Singh, P. Recent Advancements in End-to-End Autonomous Driving using Deep Learning: A Survey. *arXiv* **2023**, arXiv:2307.04370.
16. Ganesan, M.; Kandhasamy, S.; Chokkalingam, B.; Mihet-Popa, L. A Comprehensive Review on Deep Learning-Based Motion Planning and End-to-End Learning for Self-Driving Vehicle. *IEEE Access* **2024**, *12*, 66031–66067. [[CrossRef](#)]

17. Zhao, J.; Zhao, W.; Deng, B.; Wang, Z.; Zhang, F.; Zheng, W.; Cao, W.; Nan, J.; Lian, Y.; Burke, A.F. Autonomous driving system: A comprehensive survey. *Expert Syst. Appl.* **2024**, *242*, 122836. [\[CrossRef\]](#)
18. Hu, Y.; Hu, L.; Kong, Q.; Fan, B. A Survey on End-to-end Perception and Prediction for Autonomous Driving. *Mach. Intell. Res.* **2025**, *22*, 999–1030. [\[CrossRef\]](#)
19. Zhao, J.; Wu, Y.; Deng, R.; Xu, S.; Gao, J.; Burke, A. A Survey of Autonomous Driving from a Deep Learning Perspective. *ACM Comput. Surv.* **2025**, *57*, 1–60. [\[CrossRef\]](#)
20. Yang, Y.; Han, C.; Mao, R.; Wang, H.; Chen, Z.; Yang, Y.; Ma, Q.; Chen, X.; Shi, S.; Zhang, Z. Survey of General End-to-End Autonomous Driving: A Unified Perspective. *TechRxiv* **2025**. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Dong, W.; Lu, S.; Chen, X.; Zhang, S.; Liu, Q.; Liu, Z.; Chen, L.; Wang, H.; Cai, Y. End-to-End Autonomous Driving: From Classic Paradigm to Large Model Empowerment: A Comprehensive Survey. *IEEE Internet Things J.* **2026**, *13*, 3870–3898. [\[CrossRef\]](#)
22. Xiao, Y.; Codevilla, F.; Gurram, A.; Urfalioglu, O.; López, A.M. Multimodal End-to-End Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 537–547. [\[CrossRef\]](#)
23. Hang, P.; Lv, C.; Xing, Y.; Huang, C.; Hu, Z. Human-like Decision Making for Autonomous Driving: A Noncooperative Game Theoretic Approach. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 2076–2087. [\[CrossRef\]](#)
24. Schwarting, W.; Alonso-Mora, J.; Rus, D. Planning and Decision-Making for Autonomous Vehicles. *Annu. Rev. Control Robot. Auton. Syst.* **2018**, *1*, 187–210. [\[CrossRef\]](#)
25. Sharma, O.; Sahoo, N.C.; Puhan, N.B. Dynamic Planning of Optimally Safe Lane-change Trajectory for Autonomous Driving on Multi-lane Highways Using a Fuzzy Logic-based Collision Estimator. *ACM J. Auton. Transp. Syst.* **2024**, *1*, 1–50. [\[CrossRef\]](#)
26. Batiha, I.M.; Ababneh, O.Y.; Al-Nana, A.A.; Alshanti, W.G.; Alshorm, S.; Momani, S. A Numerical Implementation of Fractional-Order PID Controllers for Autonomous Vehicles. *Axioms* **2023**, *12*, 306. [\[CrossRef\]](#)
27. Chen, Z.; Lai, J.; Li, P.; Awad, O.I.; Zhu, Y. Prediction Horizon-Varying Model Predictive Control (MPC) for Autonomous Vehicle Control. *Electronics* **2024**, *13*, 1442. [\[CrossRef\]](#)
28. Pomerleau, D.A. ALVINN: An Autonomous Land Vehicle in a Neural Network. In Proceedings of the Advances in Neural Information Processing Systems 1 (NIPS 1988), Denver, CO, USA, 28 November–1 December 1988; pp. 305–313.
29. Paniego, S.; Calvo-Palomino, R.; Cañas, J. Enhancing end-to-end control in autonomous driving through kinematic-infused and visual memory imitation learning. *Neurocomputing* **2024**, *600*, 128161. [\[CrossRef\]](#)
30. Dong, Z.; Zhu, Y.; Li, Y.; Mahon, K.; Sun, Y. Generalizing End-To-End Autonomous Driving In Real-World Environments Using Zero-Shot LLMs. *arXiv* **2024**, arXiv:2411.14256.
31. Hu, S.; Chen, L.; Wu, P.; Li, H.; Yan, J.; Tao, D. ST-P3: End-to-End Vision-Based Autonomous Driving via Spatial-Temporal Feature Learning. In Proceedings of the Computer Vision—ECCV 2022; Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T., Eds.; Lecture Notes in Computer Science; Springer Nature: Cham, Switzerland, 2022; Volume 13698, pp. 533–549.
32. Hu, A.; Murez, Z.; Mohan, N.; Dudas, S.; Hawke, J.; Badrinarayanan, V.; Cipolla, R.; Kendall, A. FIERY: Future Instance Prediction in Bird’s-Eye View from Surround Monocular Cameras. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 15273–15282.
33. Sun, W.; Lin, X.; Shi, Y.; Zhang, C.; Wu, H.; Zheng, S. SparseDrive: End-to-End Autonomous Driving via Sparse Scene Representation. *arXiv* **2024**, arXiv:2405.19620.
34. Song, Z.; Jia, C.; Liu, L.; Pan, H.; Zhang, Y.; Wang, J.; Zhang, X.; Xu, S.; Yang, L.; Luo, Y. Don’t Shake the Wheel: Momentum-Aware Planning in End-to-End Autonomous Driving. *arXiv* **2025**, arXiv:2503.03125.
35. Bansal, M.; Krizhevsky, A.; Ogale, A. ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst. In Proceedings of the Robotics: Science and Systems XV, Robotics: Science and Systems Foundation, Freiburg, Germany, 22–26 June 2019.
36. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
37. Shao, H.; Wang, L.; Chen, R.; Waslander, S.L.; Li, H.; Liu, Y. ReasonNet: End-to-End Driving with Temporal and Global Reasoning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 13723–13733. Available online: https://openaccess.thecvf.com/content/CVPR2023/html/Shao_ReasonNet_End-to-End_Driving_With_Temporal_and_Global_Reasoning_CVPR_2023_paper.html (accessed on 22 July 2025).
38. Peng, L.; Chen, Z.; Fu, Z.; Liang, P.; Cheng, E. BEVSegFormer: Bird’s Eye View Semantic Segmentation from Arbitrary Camera Rigs. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 5935–5943. Available online: https://openaccess.thecvf.com/content/WACV2023/html/Peng_BEVSegFormer_Birds_Eye_View_Semantic_Segmentation_From_Arbitrary_Camera_Rigs_WACV_2023_paper.html (accessed on 13 January 2025).
39. Hanselmann, N.; Renz, K.; Chitta, K.; Bhattacharyya, A.; Geiger, A. KING: Generating Safety-Critical Driving Scenarios for Robust Imitation via Kinematics Gradients. In Proceedings of the Computer Vision—ECCV 2022; Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T., Eds.; Lecture Notes in Computer Science; Springer Nature: Cham, Switzerland, 2022; Volume 13698, pp. 335–352.

40. Osiński, B.; Jakubowski, A.; Miłoś, P.; Zięcina, P.; Galias, C.; Homoceanu, S.; Michalewski, H. Simulation-based Reinforcement Learning for Real-world Autonomous Driving. *arXiv* **2019**, arXiv:1911.12905.
41. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)] [[PubMed](#)]
42. Bijelic, M.; Gruber, T.; Ritter, W. Benchmarking Image Sensors Under Adverse Weather Conditions for Autonomous Driving. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1773–1779.
43. Goberville, N.; El-Yabroudi, M.Z.; Omwanas, M.; Rojas, J.; Meyer, R.; Asher, Z.; Abdel-Qader, I. Analysis of LiDAR and Camera Data in Real-World Weather Conditions for Autonomous Vehicle Operations. *SAE Int. J. Adv. Curr. Pract. Mobil.* **2020**, *2*, 2428–2434. [[CrossRef](#)]
44. Zeng, W.; Wang, S.; Liao, R.; Chen, Y.; Yang, B.; Urtasun, R. DSDNet: Deep Structured Self-driving Network. In *Proceedings of the Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2020; Volume 12366, pp. 156–172.
45. Cai, P.; Wang, S.; Wang, H.; Liu, M. Carl-Lead: LiDAR-based End-to-End Autonomous Driving with Contrastive Deep Reinforcement Learning. *arXiv* **2021**, arXiv:2109.08473.
46. Hu, P.; Huang, A.; Dolan, J.; Held, D.; Ramanan, D. Safe Local Motion Planning with Self-Supervised Freespace Forecasting. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 12727–12736.
47. Simon, M.; Milz, S.; Amende, K.; Gross, H.-M. Complex-YOLO: An Euler-Region-Proposal for Real-Time 3D Object Detection on Point Clouds. In *Proceedings of the Computer Vision—ECCV 2018 Workshops*; Leal-Taixé, L., Roth, S., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2019; Volume 11129, pp. 197–209.
48. Chen, D.; Zhou, B.; Koltun, V.; Krähenbühl, P. Learning by Cheating. *arXiv* **2019**, arXiv:1912.12294.
49. Liu, R.; Wang, X.; Wang, W.; Yang, Y. Bird’s-Eye-View Scene Graph for Vision-Language Navigation. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 10934–10946.
50. Wu, H.; Zhang, Z.; Lin, S.; Qin, T.; Pan, J.; Zhao, Q.; Xu, C.; Yang, M. BLOS-BEV: Navigation Map Enhanced Lane Segmentation Network, Beyond Line of Sight. In Proceedings of the 2024 IEEE Intelligent Vehicles Symposium (IV), Jeju Island, Republic of Korea, 2–5 June 2024; pp. 3212–3219.
51. Jiang, H.; Chang, L.; Li, Q.; Chen, D. Deep Transfer Learning Enable End-to-End Steering Angles Prediction for Self-driving Car. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020.
52. Wu, T.; Luo, A.; Huang, R.; Cheng, H.; Zhao, Y. End-to-End Driving Model for Steering Control of Autonomous Vehicles with Future Spatiotemporal Features. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019.
53. Valiente, R.; Zaman, M.; Ozer, S.; Fallah, Y.P. Controlling Steering Angle for Cooperative Self-driving Vehicles utilizing CNN and LSTM-based Deep Networks. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019.
54. Chang, B.R.; Tsai, H.-F.; Chang, F.-Y. Boosting the Response of Object Detection and Steering Angle Prediction for Self-Driving Control. *Electronics* **2023**, *12*, 4281. [[CrossRef](#)]
55. Navarro, P.J.; Miller, L.; Rosique, F.; Fernández-Isla, C.; Gila-Navarro, A. End-to-End Deep Neural Network Architectures for Speed and Steering Wheel Angle Prediction in Autonomous Driving. *Electronics* **2021**, *10*, 1266. [[CrossRef](#)]
56. Zeng, W.; Luo, W.; Su, S.; Sadat, A.; Yang, B.; Casas, S.; Urtasun, R. End-To-End Interpretable Neural Motion Planner. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8652–8661.
57. Chitta, K.; Prakash, A.; Jaeger, B.; Yu, Z.; Renz, K.; Geiger, A. TransFuser: Imitation with Transformer-Based Sensor Fusion for Autonomous Driving. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 12878–12895. [[CrossRef](#)] [[PubMed](#)]
58. Kendall, A.; Hawke, J.; Janz, D.M.; Mazur, P.; Reda, D.; Allen, J.-M.; Lam, V.-D.; Bewley, A.; Shah, A. Learning to Drive in a Day. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 8248–8254.
59. Agand, P.; Mahdavian, M.; Savva, M.; Chen, M. LeTFuser: Light-weight End-to-end Transformer-Based Sensor Fusion for Autonomous Driving with Multi-Task Learning. *arXiv* **2023**, arXiv:2310.13135.
60. Wang, P.; Zhu, M.; Zheng, X.; Lu, H.; Zhong, H.; Chen, X.; Shen, S.; Wang, X.; Wang, Y.; Wang, F.-Y. BEVGPT: Generative Pre-trained Foundation Model for Autonomous Driving Prediction, Decision-Making, and Planning. *IEEE Trans. Intell. Veh.* **2024**, 1–13. [[CrossRef](#)]
61. Codevilla, F.; Santana, E.; Lopez, A.; Gaidon, A. Exploring the Limitations of Behavior Cloning for Autonomous Driving. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9328–9337.
62. Chen, S.-F.; Wang, H.-C.; Hsu, M.-H.; Lai, C.-M.; Sun, S.-H. Diffusion Model-Augmented Behavioral Cloning. *arXiv* **2024**, arXiv:2302.13335.

63. Li, G.; Ji, Z.; Li, S.; Luo, X.; Qu, X. Driver Behavioral Cloning for Route Following in Autonomous Vehicles Using Task Knowledge Distillation. *IEEE Trans. Intell. Veh.* **2023**, *8*, 1025–1033. [CrossRef]
64. Ross, S.; Gordon, G.; Bagnell, D. A Reduction of Imitation Learning and Structured Prediction to No-Regret On-line Learning. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, Fort Lauderdale, FL, USA, 14 June 2011; pp. 627–635. Available online: <https://proceedings.mlr.press/v15/ross11a.html> (accessed on 13 January 2025).
65. Prakash, A.; Behl, A.; Ohn-Bar, E.; Chitta, K.; Geiger, A. Exploring Data Aggregation in Policy Learning for Vision-Based Urban Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11763–11773. Available online: https://openaccess.thecvf.com/content_CVPR_2020/html/Prakash_Exploring_Data_Aggregation_in_Policy_Learning_for_Vision-Based_Urban_Autonomous_CVPR_2020_paper.html (accessed on 13 January 2025).
66. Zhang, J.; Cho, K. Query-Efficient Imitation Learning for End-to-End Simulated Driving. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
67. Adams, S.; Cody, T.; Beling, P.A. A survey of inverse reinforcement learning. *Artif. Intell. Rev.* **2022**, *55*, 4307–4346. [CrossRef]
68. Zhou, Z.; Chen, J. Modeling Driver Lane Change Behavior Using Inverse Reinforcement Learning. In Proceedings of the 2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI), Mt. Pleasant, MI, USA, 13–14 April 2024; pp. 1–5.
69. Liu, J.; Boyle, L.N.; Banerjee, A.G. An Inverse Reinforcement Learning Approach for Customizing Automated Lane Change Systems. *IEEE Trans. Veh. Technol.* **2022**, *71*, 9261–9271. [CrossRef]
70. Wu, Z.; Sun, L.; Zhan, W.; Yang, C.; Tomizuka, M. Efficient Sampling-Based Maximum Entropy Inverse Reinforcement Learning with Application to Autonomous Driving. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5355–5362. [CrossRef]
71. Sun, Y.; Chu, Y.; Xu, T.; Li, J.; Ji, X. Inverse Reinforcement Learning Based: Segmented Lane-Change Trajectory Planning with Consideration of Interactive Driving Intention. *IEEE Trans. Veh. Technol.* **2022**, *71*, 11395–11407. [CrossRef]
72. Ho, J.; Ermon, S. Generative Adversarial Imitation Learning. *arXiv* **2016**, arXiv:1606.03476.
73. Li, Y.; Song, J.; Ermon, S. InfoGAIL: Interpretable imitation learning from visual demonstrations. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 3815–3825.
74. Lee, G.; Kim, D.; Oh, W.; Lee, K.; Oh, S. MixGAIL: Autonomous Driving Using Demonstrations with Mixed Qualities. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January, 2020; pp. 5425–5430.
75. Jaeger, B.; Geiger, A. An Invitation to Deep Reinforcement Learning. *arXiv* **2024**, arXiv:2312.08365.
76. Toromanoff, M.; Wirbel, E.; Moutarde, F. End-to-End Model-Free Reinforcement Learning for Urban Driving Using Implicit Affordances. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7153–7162. Available online: https://openaccess.thecvf.com/content_CVPR_2020/html/Toromanoff_End-to-End_Model-Free_Reinforcement_Learning_for_Urban_Driving_Using_Implicit_Affordances_CVPR_2020_paper.html (accessed on 25 November 2024).
77. Ahmed, M.; Lim, C.P.; Nahavandi, S. A Deep Q-Network Reinforcement Learning-Based Model for Autonomous Driving. In Proceedings of the 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Melbourne, Australia, 17–20 October 2021; pp. 739–744.
78. Haritz, P.; Wanke, D.; Liebig, T. Enhancing Safety for Autonomous Agents in Partly Concealed Urban Traffic Environments Through Representation-Based Shielding. *arXiv* **2024**, arXiv:2407.04343.
79. Peng, B.; Sun, Q.; Li, S.E.; Kum, D.; Yin, Y.; Wei, J.; Gu, T. End-to-End Autonomous Driving Through Dueling Double Deep Q-Network. *Automot. Innov.* **2021**, *4*, 328–337. [CrossRef]
80. Cai, P.; Wang, H.; Sun, Y.; Liu, M. DQ-GAT: Towards Safe and Efficient Autonomous Driving with Deep Q-Learning and Graph Attention Networks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21102–21112. [CrossRef]
81. Anzalone, L.; Barra, P.; Barra, S.; Castiglione, A.; Nappi, M. An End-to-End Curriculum Learning Approach for Autonomous Driving Scenarios. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 19817–19826. [CrossRef]
82. Zhao, J.; Zhao, Y.; Li, W.; Zeng, C. End-to-End Autonomous Driving Algorithm Based on PPO and Its Implementation. In Proceedings of the 2024 IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS), Kaifeng, China, 17–19 May 2024; pp. 1852–1858.
83. Siboo, S.; Bhattacharyya, A.; Raj, R.N.; Ashwin, S.H. An Empirical Study of DDPG and PPO-Based Reinforcement Learning Algorithms for Autonomous Driving. *IEEE Access* **2023**, *11*, 125094–125108. [CrossRef]
84. Zou, Q.; Xiong, K.; Hou, Y. An end-to-end learning of driving strategies based on DDPG and imitation learning. In Proceedings of the 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 3190–3195.

85. Song, W.; Liu, S.; Li, Y.; Yang, Y.; Xiang, C. Smooth Actor-Critic Algorithm for End-to-End Autonomous Driving. In Proceedings of the 2020 American Control Conference (ACC), Online, 1–3 July 2020; pp. 3242–3248. [CrossRef]
86. Lin, X.; Huang, J.; Zhang, B.; Zhou, B.; Chen, Z. A velocity adaptive steering control strategy of autonomous vehicle based on double deep Q-learning network with varied agents. *Eng. Appl. Artif. Intell.* **2025**, *139*, 109655. [CrossRef]
87. Wang, Z.; Schaul, T.; Hessel, M.; van Hasselt, H.; Lanctot, M.; de Freitas, N. Dueling Network Architectures for Deep Reinforcement Learning. In Proceedings of the 33rd International Conference on Machine Learning (ICML), New York, NY, USA, 19–24 June 2016; pp. 1995–2003.
88. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous Methods for Deep Reinforcement Learning. In Proceedings of the 33rd International Conference on Machine Learning, PMLR, New York, NY, USA, 20–22 June 2016; pp. 1928–1937.
89. Singh, A. End-to-end Autonomous Driving using Deep Learning: A Systematic Review. *arXiv* **2023**, arXiv:2311.18636.
90. Wang, Y.; Wang, J.; Yang, Y.; Li, Z.; Zhao, X. An End-to-End Deep Reinforcement Learning Model Based on Proximal Policy Optimization Algorithm for Autonomous Driving of Off-Road Vehicle. In *Proceedings of the 2022 International Conference on Autonomous Unmanned Systems (ICAUS 2022)*; Fu, W., Gu, M., Niu, Y., Eds.; Springer Nature: Singapore, 2023; pp. 2692–2704.
91. Chen, S.; Wang, M.; Song, W.; Yang, Y.; Li, Y.; Fu, M. Stabilization Approaches for Reinforcement Learning-Based End-to-End Autonomous Driving. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4740–4750. [CrossRef]
92. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; pp. 1861–1870.
93. Uppuluri, B.; Patel, A.; Mehta, N.; Kamath, S.; Chakraborty, P. CuRLA: Curriculum Learning Based Deep Reinforcement Learning for Autonomous Driving. *arXiv* **2025**, arXiv:2501.04982.
94. Rizehvandi, A.; Azadi, S.; Eichberger, A. Decision-Making Policy for Autonomous Vehicles on Highways Using Deep Reinforcement Learning (DRL) Method. *Automation* **2024**, *5*, 564–577. [CrossRef]
95. Chu, D.; Bai, L.; Huang, J.; Fang, Z.; Zhang, P.; Kang, W.; Lin, H. Enhanced Safety in Autonomous Driving: Integrating Latent State Diffusion Model for End-to-End Navigation. *arXiv* **2024**, arXiv:2407.06317.
96. Zuo, J.; Hu, H.; Zhou, Z.; Cui, Y.; Liu, Z.; Wang, J.; Guan, N.; Wang, J.; Xue, C.J. RALAD: Bridging the Real-to-Sim Domain Gap in Autonomous Driving with Retrieval-Augmented Learning. *arXiv* **2025**, arXiv:2501.12296.
97. Kamath, U.; Liu, J.; Whitaker, J. Transfer Learning: Domain Adaptation. In *Proceedings of the Deep Learning for NLP and Speech Recognition*; Kamath, U., Liu, J., Whitaker, J., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 495–535.
98. Munir, F.; Azam, S.; Jeon, M. SSTN: Self-Supervised Domain Adaptation Thermal Object Detection for Autonomous Driving. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech, 27 September–1 October 2021; pp. 206–213.
99. Xu, J.; Xiao, L.; López, A.M. Self-Supervised Domain Adaptation for Computer Vision Tasks. *IEEE Access* **2019**, *7*, 156694–156706. [CrossRef]
100. Achituve, I.; Maron, H.; Chechik, G. Self-Supervised Learning for Domain Adaptation on Point Clouds. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 123–133. Available online: https://openaccess.thecvf.com/content/WACV2021/html/Achituve_Self-Supervised_Learning_for_Domain_Adaptation_on_Point_Clouds_WACV_2021_paper.html (accessed on 13 January 2025).
101. Yang, L.; Liang, X.; Wang, T.; Xing, E. Real-to-Virtual Domain Unification for End-to-End Autonomous Driving. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 530–545. Available online: https://openaccess.thecvf.com/content_ECCV_2018/html/Luona_Yang_Real-to-Virtual_Domain_Uni_ECCV_2018_paper.html (accessed on 1 March 2025).
102. Wozniak, M.K.; Hansson, M.; Thiel, M.; Jensfelt, P. UADA3D: Unsupervised Adversarial Domain Adaptation for 3D Object Detection with Sparse LiDAR and Large Domain Gaps. *arXiv* **2024**, arXiv:2403.17633.
103. Zhao, S.; Wang, Y.; Li, B.; Wu, B.; Gao, Y.; Xu, P.; Darrell, T.; Keutzer, K. ePointDA: An End-to-End Simulation-to-Real Domain Adaptation Framework for LiDAR Point Cloud Segmentation. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 3500–3509. [CrossRef]
104. Hu, C.; Hudson, S.; Ethier, M.; Al-Sharman, M.; Rayside, D.; Melek, W. Sim-to-Real Domain Adaptation for Lane Detection and Classification in Autonomous Driving. In Proceedings of the 2022 IEEE Intelligent Vehicles Symposium (IV), Aachen, Germany, 4–9 June 2022; pp. 457–463.
105. Rao, K.; Harris, C.; Irpan, A.; Levine, S.; Ibarz, J.; Khansari, M. RL-CycleGAN: Reinforcement Learning Aware Simulation-to-Real. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11154–11163.
106. Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Yu, Q.; Dai, J. BEVFormer: Learning Bird’s-Eye-View Representation From LiDAR-Camera via Spatiotemporal Transformers. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 2020–2036. [CrossRef]

107. Nayakanti, N.; Al-Rfou, R.; Zhou, A.; Goel, K.; Refaat, K.S.; Sapp, B. Wayformer: Motion Forecasting via Simple & Efficient Attention Networks. *arXiv* **2022**, arXiv:2207.05844.
108. Zheng, W.; Song, R.; Guo, X.; Zhang, C.; Chen, L. GenAD: Generative End-to-End Autonomous Driving. *arXiv* **2024**, arXiv:2402.11502.
109. Jia, X.; You, J.; Zhang, Z.; Yan, J. DriveTransformer: Unified Transformer for Scalable End-to-End Autonomous Driving. In Proceedings of the Thirteenth International Conference on Learning Representations (ICLR 2025), Singapore, 24–28 April 2025. Available online: <https://openreview.net/forum?id=M42KR4W9P5> (accessed on 22 March 2025).
110. Liao, B.; Chen, S.; Yin, H.; Jiang, B.; Wang, C.; Yan, S.; Zhang, X.; Li, X.; Zhang, Y.; Zhang, Q.; et al. DiffusionDrive: Truncated Diffusion Model for End-to-End Autonomous Driving. *arXiv* **2024**, arXiv:2411.15139.
111. Jiang, X.; Ma, Y.; Li, P.; Xu, L.; Wen, X.; Zhan, K.; Xia, Z.; Jia, P.; Lang, X.; Sun, S. TransDiffuser: Diverse Trajectory Generation with Decorrelated Multi-modal Representation for End-to-End Autonomous Driving. *arXiv* **2025**, arXiv:2505.09315.
112. Guan, Y.; Liao, H.; Wang, C.; Liu, X.; Zhang, J.; Li, Z. World Model-Based End-to-End Scene Generation for Accident Anticipation in Autonomous Driving. *arXiv* **2025**, arXiv:2507.12762.
113. Hu, A.; Russell, L.; Yeo, H.; Murez, Z.; Fedoseev, G.; Kendall, A.; Shotton, J.; Corrado, G. GAIA-1: A Generative World Model for Autonomous Driving. *arXiv* **2023**, arXiv:2309.17080.
114. Wang, X.; Zhu, Z.; Huang, G.; Chen, X.; Zhu, J.; Lu, J. DriveDreamer: Towards Real-world-driven World Models for Autonomous Driving. *arXiv* **2023**, arXiv:2309.09777.
115. Zhao, G.; Wang, X.; Zhu, Z.; Chen, X.; Huang, G.; Bao, X.; Wang, X. DriveDreamer-2: LLM-Enhanced World Models for Diverse Driving Video Generation. *arXiv* **2024**, arXiv:2403.06845.
116. Min, C.; Zhao, D.; Xiao, L.; Zhao, J.; Xu, X.; Zhu, Z.; Jin, L.; Li, J.; Guo, Y.; Xing, J.; et al. DriveWorld: 4D Pre-trained Scene Understanding via World Models for Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–24 June 2024; pp. 15522–15533. Available online: https://openaccess.thecvf.com/content/CVPR2024/html/Min_DriveWorld_4D_Pre-trained_Scene_Understanding_via_World_Models_for_Autonomous_CVPR_2024_paper.html (accessed on 1 January 2025).
117. Min, C.; Zhao, D.; Xiao, L.; Nie, Y.; Dai, B. UniWorld: Autonomous Driving Pre-training via World Models. *arXiv* **2023**, arXiv:2308.07234.
118. Zhao, G.; Ni, C.; Wang, X.; Zhu, Z.; Zhang, X.; Wang, Y.; Huang, G.; Chen, X.; Wang, B.; Zhang, Y.; et al. DriveDreamer4D: World Models Are Effective Data Machines for 4D Driving Scene Representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 11–15 June 2025; pp. 12015–12026. Available online: https://openaccess.thecvf.com/content/CVPR2025/html/Zhao_DriveDreamer4D_World_Models_Are_Effective_Data_Machines_for_4D_Driving_CVPR_2025_paper.html (accessed on 25 July 2025).
119. Yang, Z.; Guo, X.; Ding, C.; Wang, C.; Wu, W. Physical Informed Driving World Model. *arXiv* **2024**, arXiv:2412.08410.
120. Bogdoll, D.; Yang, Y.; Joseph, T.; Yazgan, M.; Zöllner, J.M. MUVO: A Multimodal Generative World Model for Autonomous Driving with Geometric Representations. *arXiv* **2023**, arXiv:2311.11762.
121. Wang, Y.; He, J.; Fan, L.; Li, H.; Chen, Y.; Zhang, Z. Driving into the Future: Multiview Visual Forecasting and Planning with World Model for Autonomous Driving. *arXiv* **2023**, arXiv:2311.17918.
122. Yang, Y.; Mei, J.; Ma, Y.; Du, S.; Chen, W.; Qian, Y.; Feng, Y.; Liu, Y. Driving in the Occupancy World: Vision-Centric 4D Occupancy Forecasting and Planning via World Models for Autonomous Driving. *arXiv* **2024**, arXiv:2408.14197.
123. Eykholt, K.; Evtimov, I.; Fernandes, E.; Li, B.; Rahmati, A.; Xiao, C.; Prakash, A.; Kohno, T.; Song, D. Robust Physical-World Attacks on Deep Learning Visual Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1625–1634.
124. Xu, K.; Zhang, G.; Liu, S.; Fan, Q.; Sun, M.; Chen, H.; Chen, P.-Y.; Wang, Y.; Lin, X. Adversarial T-Shirt! Evading Person Detectors in a Physical World. In *Proceedings of the Computer Vision—ECCV 2020*; Springer Nature: Cham, Switzerland, 2020; pp. 665–681.
125. Knox, W.B.; Allievi, A.; Banzhaf, H.; Schmitt, F.; Stone, P. Reward (Mis)design for autonomous driving. *Artif. Intell.* **2023**, *316*, 103829. [CrossRef]
126. Zhang, J.; Lou, Y.; Wang, J.; Wu, K.; Lu, K.; Jia, X. Evaluating Adversarial Attacks on Driving Safety in Vision-Based Autonomous Vehicles. *IEEE Internet Things J.* **2022**, *9*, 3443–3456. [CrossRef]
127. Zhu, M.; Sidhu, A.; Redmill, K.A. Training Adversarial Yet Safe Agent to Characterize the Safety Performance of Highly Automated Vehicles. In Proceedings of the 2024 IEEE International Automated Vehicle Validation Conference (IAVVC), Pittsburgh, PA, USA, 21–23 October 2024; pp. 1–7.
128. Lindemann, L.; Zhao, Y.; Yu, X.; Pappas, G.J.; Deshmukh, J.V. Formal Verification and Control with Conformal Prediction. *arXiv* **2024**, arXiv:2409.00536.
129. Hu, X.; Chen, P.; Wen, Y.; Tang, B.; Chen, L. Long and Short-Term Constraints Driven Safe Reinforcement Learning for Autonomous Driving. *arXiv* **2024**, arXiv:2403.18209.

130. Gharavi, L.; Dabiri, A.; Verkuijlen, J.; Schutter, B.D.; Baldi, S. Proactive Emergency Collision Avoidance for Automated Driving in Highway Scenarios. *IEEE Trans. Control Syst. Technol.* **2024**, *33*, 1164–1177. [\[CrossRef\]](#)
131. Shang, X.; Eskandarian, A. Emergency Collision Avoidance and Mitigation Using Model Predictive Control and Artificial Potential Function. *IEEE Trans. Intell. Veh.* **2023**, *8*, 3458–3472. [\[CrossRef\]](#)
132. He, X.; Lou, B.; Yang, H.; Lv, C. Robust Decision Making for Autonomous Vehicles at Highway On-Ramps: A Constrained Adversarial Reinforcement Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 4103–4113. [\[CrossRef\]](#)
133. Zhang, Z.; Liu, Q.; Li, Y.; Lin, K.; Li, L. Safe Reinforcement Learning in Autonomous Driving with Epistemic Uncertainty Estimation. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 13653–13666. [\[CrossRef\]](#)
134. Gao, C.; Wang, G.; Shi, W.; Wang, Z.; Chen, Y. Autonomous Driving Security: State of the Art and Challenges. *IEEE Internet Things J.* **2022**, *9*, 7572–7595. [\[CrossRef\]](#)
135. Lan, G.; Hao, Q. End-To-End Planning of Autonomous Driving in Industry and Academia: 2022–2023. *arXiv* **2023**, arXiv:2401.08658.
136. Li, Y.; Fan, L.; He, J.; Wang, Y.; Chen, Y.; Zhang, Z.; Tan, T. Enhancing End-to-End Autonomous Driving with Latent World Model. *arXiv* **2024**, arXiv:2406.08481.
137. Nakamura, K.; Peters, L.; Bajcsy, A. Generalizing Safety Beyond Collision-Avoidance via Latent-Space Reachability Analysis. *arXiv* **2025**, arXiv:2502.00935.
138. Omeiza, D.; Webb, H.; Jirotko, M.; Kunze, L. Explanations in Autonomous Driving: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 10142–10162. [\[CrossRef\]](#)
139. Zablocki, É.; Ben-Younes, H.; Pérez, P.; Cord, M. Explainability of Deep Vision-Based Autonomous Driving Systems: Review and Challenges. *Int. J. Comput. Vis.* **2022**, *130*, 2425–2452. [\[CrossRef\]](#)
140. Atakishiyev, S.; Salameh, M.; Yao, H.; Goebel, R. Explainable Artificial Intelligence for Autonomous Driving: A Comprehensive Overview and Field Guide for Future Research Directions. *IEEE Access* **2024**, *12*, 101603–101625. [\[CrossRef\]](#)
141. Li, J.; Li, H.; Liu, J.; Zou, Z.; Ye, X.; Wang, F.; Huang, J.; Wu, H.; Wang, H. Exploring the Causality of End-to-End Autonomous Driving. *arXiv* **2024**, arXiv:2407.06546.
142. Melis, D.A.; Jaakkola, T. Towards Robust Interpretability with Self-Explaining Neural Networks. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2018. Available online: https://proceedings.neurips.cc/paper_files/paper/2018/hash/3e9f0fc9b2f89e043bc6233994dfcf76-Abstract.html (accessed on 4 November 2024).
143. Wang, C.; Aouf, N. Explainable Deep Adversarial Reinforcement Learning Approach for Robust Autonomous Driving. *IEEE Trans. Intell. Veh.* **2024**, *10*, 2551–2563. [\[CrossRef\]](#)
144. Lateef, F.; Kas, M.; Ruichek, Y. Saliency Heat-Map as Visual Attention for Autonomous Driving Using Generative Adversarial Network (GAN). *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 5360–5373. [\[CrossRef\]](#)
145. Mohseni, S.; Jagadeesh, A.; Wang, Z. Predicting Model Failure using Saliency Maps in Autonomous Driving Systems. *arXiv* **2019**, arXiv:1905.07679.
146. Liu, Q.; Tang, Y.; Li, X.; Yang, F.; Wang, K.; Li, Z. MV-STGHAT: Multi-View Spatial-Temporal Graph Hybrid Attention Network for Decision-Making of Connected and Autonomous Vehicles. *IEEE Trans. Veh. Technol.* **2024**, *74*, 5594–5609.
147. Wang, X.; Wu, Z.; Jin, B.; Lin, M.; Zou, F.; Liao, L. MDSTF: A multi-dimensional spatio-temporal feature fusion trajectory prediction model for autonomous driving. *Complex Intell. Syst.* **2024**, *10*, 6647–6665. [\[CrossRef\]](#)
148. Wu, J.; Huang, Z.; Hu, Z.; Lv, C. Toward Human-in-the-Loop AI: Enhancing Deep Reinforcement Learning via Real-Time Human Guidance for Autonomous Driving. *Engineering* **2023**, *21*, 75–91. [\[CrossRef\]](#)
149. Wu, Z.; Geiger, A.; Icard, T.; Potts, C.; Goodman, N. Interpretability at Scale: Identifying Causal Mechanisms in Alpaca. In *Proceedings of the Advances in Neural Information Processing Systems*, New Orleans, LA, USA, 10–16 December 2023; Volume 36, pp. 78205–78226.
150. Caton, S.; Haas, C. Fairness in Machine Learning: A Survey. *ACM Comput. Surv.* **2024**, *56*, 1–38. [\[CrossRef\]](#)
151. Feng, S.; Sun, H.; Yan, X.; Zhu, H.; Zou, Z.; Shen, S.; Liu, H.X. Dense Reinforcement Learning for Safety Validation of Autonomous Vehicles. *Nature* **2023**, *615*, 620–627. [\[CrossRef\]](#) [\[PubMed\]](#)
152. Zhao, T.; Yurtsever, E.; Paulson, J.A.; Rizzoni, G. Formal Certification Methods for Automated Vehicle Safety Assessment. *IEEE Trans. Intell. Veh.* **2023**, *8*, 232–249. [\[CrossRef\]](#)
153. Xu, Z.; Zhang, Y.; Xie, E.; Zhao, Z.; Guo, Y.; Wong, K.K.; Li, Z.; Zhao, H. DriveGPT4: Interpretable End-to-End Autonomous Driving via Large Language Model. *IEEE Robot. Autom. Lett.* **2024**, *9*, 8186–8193. [\[CrossRef\]](#)
154. Hu, Y.; Jia, X.; Tomizuka, M.; Zhan, W. Causal-based Time Series Domain Generalization for Vehicle Intention Prediction. *arXiv* **2021**, arXiv:2112.02093.
155. Liu, Q.; Lv, J.; Zhu, Y. Prediction of Surrounding Vehicle Trajectories for Autonomous Driving Based on Counterfactual Causal-Reasoning Graph Neural Network. *IEEE Trans. Intell. Veh.* **2024**, 1–16. [\[CrossRef\]](#)

156. Pourkeshavarz, M.; Zhang, J.; Rasouli, A. CaDeT: A Causal Disentanglement Approach for Robust Trajectory Prediction in Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 14874–14884. Available online: https://openaccess.thecvf.com/content/CVPR2024/html/Pourkeshavarz_CaDeT_a_Causal_Disentanglement_Approach_for_Robust_Trajectory_Prediction_in_CVPR_2024_paper.html (accessed on 6 August 2025).
157. Giamattei, L.; Guerriero, A.; Pietrantuono, R.; Russo, S. Causality-driven Testing of Autonomous Driving Systems. *ACM Trans. Softw. Eng. Methodol.* **2024**, *33*, 1–35. [CrossRef]
158. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
159. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2636–2645.
160. Huang, X.; Cheng, X.; Geng, Q.; Cao, B.; Zhou, D.; Wang, P.; Lin, Y.; Yang, R. The ApolloScape Dataset for Autonomous Driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 954–960.
161. Sun, P.; Kretschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2446–2454. Available online: https://openaccess.thecvf.com/content_CVPR_2020/html/Sun_Scalability_in_Perception_for_Autonomous_Driving_Waymo_Open_Dataset_CVPR_2020_paper.html (accessed on 9 December 2024).
162. Caesar, H.; Bankiti, V.; Lang, A.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. nuScenes: A Multimodal Dataset for Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11621–11631.
163. Geyer, J.; Kassahun, Y.; Mahmudi, M.; Ricou, X.; Durgesh, R.; Chung, A.S.; Hauswald, L.; Pham, V.H.; Mühlegg, M.; Dorn, S.; et al. A2D2: Audi Autonomous Driving Dataset. *arXiv* **2020**, arXiv:2004.06320.
164. Caesar, H.; Kabzan, J.; Tan, K.S.; Fong, W.K.; Wolff, E.; Lang, A.; Fletcher, L.; Beijbom, O.; Omari, S. NuPlan: A Closed-loop ML-based Planning Benchmark for Autonomous Vehicles. *arXiv* **2022**, arXiv:2106.11810.
165. Wilson, B.; Qi, W.; Agarwal, T.; Lambert, J.; Singh, J.; Khandelwal, S.; Pan, B.; Kumar, R.; Hartnett, A.; Pontes, J.K.; et al. Argoverse 2: Next Generation Datasets for Self-Driving Perception and Forecasting. *arXiv* **2023**, arXiv:2301.00493.
166. Sima, C.; Renz, K.; Chitta, K.; Chen, L.; Zhang, H.; Xie, C.; Beißwenger, J.; Luo, P.; Geiger, A.; Li, H. DriveLM: Driving with Graph Visual Question Answering. *arXiv* **2024**, arXiv:2312.14150.
167. Alibeigi, M.; Ljungbergh, W.; Tonderski, A.; Hess, G.; Lilja, A.; Lindström, C.; Motorniuk, D.; Fu, J.; Widahl, J.; Petersson, C. Zenseact Open Dataset: A Large-scale and Diverse Multimodal Dataset for Autonomous Driving. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 4–6 October 2023; pp. 20121–20131.
168. Wang, T.; Xie, E.; Chu, R.; Li, Z.; Luo, P. DriveCoT: Integrating Chain-of-Thought Reasoning with End-to-End Driving. *arXiv* **2024**, arXiv:2403.16996.
169. Li, Y.; Li, Z.; Chen, N.; Gong, M.; Lyu, Z.; Wang, Z.; Jiang, P.; Chen, F. Multiagent Multitraversal Multimodal Self-Driving: Open MARS Dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 22041–22051.
170. Arai, H.; Miwa, K.; Sasaki, K.; Yamaguchi, Y.; Watanabe, K.; Aoki, S.; Yamamoto, I. CoVLA: Comprehensive Vision-Language-Action Dataset for Autonomous Driving. *arXiv* **2024**, arXiv:2408.10845.
171. Zhai, J.-T.; Feng, Z.; Du, J.; Mao, Y.; Liu, J.-J.; Tan, Z.; Zhang, Y.; Ye, X.; Wang, J. Rethinking the Open-Loop Evaluation of End-to-End Autonomous Driving in nuScenes. *arXiv* **2023**, arXiv:2305.10430.
172. Li, Z.; Yu, Z.; Lan, S.; Li, J.; Kautz, J.; Lu, T.; Álvarez, J.M. Is Ego Status All You Need for Open-Loop End-to-End Autonomous Driving? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 14864–14873.
173. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. In Proceedings of the 1st Annual Conference on Robot Learning, PMLR, Mountain View, CA, USA, 18 October 2017; pp. 1–16. Available online: <https://proceedings.mlr.press/v78/dosovitskiy17a.html> (accessed on 13 December 2024).
174. Shao, H.; Hu, Y.; Wang, L.; Song, G.; Waslander, S.L.; Liu, Y.; Li, H. LMDrive: Closed-Loop End-to-End Driving with Large Language Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 15120–15130.
175. Zhao, J. Simulation Research Based on Double DQN for End-to-End Autonomous Driving. In *Proceedings of the Second International Conference on Artificial Intelligence and Communication Technologies (ICAICT 2024)*; Jain, L.C., Kountcheva, R., Nakamatsu, K., Eds.; Springer Nature: Singapore, 2025; pp. 3–18.

176. Aslam, I.; Anpilogov, I.; Rausch, A. Towards Selection and Transition Between Behavior-Based Neural Networks for Automated Driving. In *Proceedings of the 2024 8th International Conference on Computer Science and Artificial Intelligence, Beijing, China, 6–8 December 2024*; Association for Computing Machinery: New York, NY, USA, 2025; pp. 340–346.
177. Lu, C.; Yue, T.; Zhang, M.; Ali, S. DeepQTest: Testing Autonomous Driving Systems with Reinforcement Learning and Real-world Weather Data. *arXiv* **2023**, arXiv:2310.05170.
178. Tesla. 2019 Impact Report. Available online: https://www.tesla.com/ns_videos/2019-tesla-impact-report.pdf (accessed on 21 July 2026).
179. XPeng. XPENG Reveals Groundbreaking Mobility Technologies at 2022 Tech Day. Available online: <https://www.xpeng.com/news/01840adee4c382358a5f2c9e246a0367> (accessed on 21 July 2026).
180. Horizon Robotics. Horizon Robotics Launches Next Generation Smart Driving Solution SuperDrive and Journey 6 Series. Available online: <https://www.horizon.auto/en/news/horizon-news/442> (accessed on 21 July 2026).
181. Li, K.; Chen, K.; Wang, H.; Hong, L.; Ye, C.; Han, J.; Chen, Y.; Zhang, W.; Xu, C.; Yeung, D.-Y.; et al. CODA: A Real-World Road Corner Case Dataset for Object Detection in Autonomous Driving. In *Proceedings of the Computer Vision—ECCV 2022*; Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T., Eds.; Springer Nature: Cham, Switzerland, 2022; pp. 406–423.
182. Russell, L.; Hu, A.; Bertoni, L.; Fedoseev, G.; Shotton, J.; Arani, E.; Corrado, G. GAIA-2: A Controllable Multi-View Generative World Model for Autonomous Driving. *arXiv* **2025**, arXiv:2503.20523.
183. Xie, J.; Zhang, Y.; Qin, Y. Is human-like decision making explainable? Towards an explainable artificial intelligence for autonomous vehicles. *Transp. Res. Interdiscip. Perspect.* **2025**, *29*, 101278. [[CrossRef](#)]
184. Lu, C.; Lu, H.; Chen, D.; Wang, H.; Li, P.; Gong, J. Human-like decision making for lane change based on the cognitive map and hierarchical reinforcement learning. *Transp. Res. Part C Emerg. Technol.* **2023**, *156*, 104328. [[CrossRef](#)]
185. Abukmeil, M.; Genovese, A.; Piuri, V.; Rundo, F.; Scotti, F. Towards Explainable Semantic Segmentation for Autonomous Driving Systems by Multi-Scale Variational Attention. In *Proceedings of the 2021 IEEE International Conference on Autonomous Systems (ICAS)*, Montréal, QC, Canada, 11–13 August 2021; pp. 1–5.
186. Khan, S.; Teeti, I.; Alitappeh, R.J.; Stoian, M.C.; Giunchiglia, E.; Singh, G.; Bradley, A.; Cuzzolin, F. ROAD-Waymo: A Large-Scale Action Awareness Dataset for Autonomous Driving. *arXiv* **2024**, arXiv:2411.01683.
187. Jiang, Y.; Mi, Q.; Zhu, L. Neurocomputational mechanism of real-time distributed learning on social networks. *Nat. Neurosci.* **2023**, *26*, 506–516. [[CrossRef](#)] [[PubMed](#)]
188. Ho, M.K.; Abel, D.; Correa, C.G.; Littman, M.L.; Cohen, J.D.; Griffiths, T.L. People construct simplified mental representations to plan. *Nature* **2022**, *606*, 129–136. [[CrossRef](#)] [[PubMed](#)]
189. Krügel, S.; Uhl, M. The risk ethics of autonomous vehicles: An empirical approach. *Sci. Rep.* **2024**, *14*, 960. [[CrossRef](#)] [[PubMed](#)]
190. Caballero, W.N.; Naveiro, R.; Insua, D.R. Modeling Ethical and Operational Preferences in Automated Driving Systems. *Decis. Anal.* **2021**, *19*, 21–43. [[CrossRef](#)]
191. Shahbaz, M.; Agarwal, S. High-Fidelity Digital Twins for Bridging the Sim2Real Gap in LiDAR-Based ITS Perception. *arXiv* **2025**, arXiv:2509.02904.
192. Samak, C.V.; Samak, T.V.; Li, B.; Krovi, V. Sim2Real Diffusion: Leveraging Foundation Vision Language Models for Adaptive Automated Driving. *IEEE Robot. Autom. Lett.* **2026**, *11*, 177–184. [[CrossRef](#)]
193. Tian, K.; Mao, J.; Zhang, Y.; Jiang, J.; Zhou, Y.; Tu, Z. NuScenes-SpatialQA: A Spatial Understanding and Reasoning Benchmark for Vision-Language Models in Autonomous Driving. *arXiv* **2025**, arXiv:2504.03164.
194. Guo, X.; Zhang, R.; Duan, Y.; He, Y.; Nie, D.; Huang, W.; Zhang, C.; Liu, S.; Zhao, H.; Chen, L. SURDS: Benchmarking Spatial Understanding and Reasoning in Driving Scenarios with Vision Language Models. *arXiv* **2025**, arXiv:2411.13112.
195. Yu, H.; Yang, W.; Zhong, J.; Yang, Z.; Fan, S.; Luo, P.; Nie, Z. End-to-End Autonomous Driving Through V2X Cooperation. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 9598–9606. [[CrossRef](#)]
196. Parada, L.; Yu, K.; Angeloudis, P. IntNet: A Communication-Driven Multi-Agent Reinforcement Learning Framework for Cooperative Autonomous Driving. *IEEE Robot. Autom. Lett.* **2025**, *10*, 2478–2485. [[CrossRef](#)]
197. Xu, R.; Chen, C.-J.; Tu, Z.; Yang, M.-H. V2X-ViT2: Improved Vision Transformers for Vehicle-to-Everything Cooperative Perception. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 650–662. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.